

corresponding to U.S. 7,080,371 B1

(19) 日本国特許庁 (J P)

(12) 公表特許公報 (A)

(11) 特許出願公表番号

特表2002-506249

(P2002-506249A)

(43) 公表日 平成14年2月26日 (2002.2.26)

(51) Int.Cl.<sup>7</sup>

G 0 6 F 9/445

識別記号

F I

G 0 6 F 9/06

データベース\* (参考)

6 1 0 Q 5 B 0 7 6

審査請求 未請求 予備審査請求 有 (全 66 頁)

(21) 出願番号 特願2000-534941(P2000-534941)  
(86) (22) 出願日 平成11年3月3日(1999.3.3)  
(85) 翻訳文提出日 平成12年9月4日(2000.9.4)  
(86) 国際出願番号 P C T / U S 9 9 / 0 4 5 8 1  
(87) 国際公開番号 W O 9 9 / 4 5 4 6 5  
(87) 国際公開日 平成11年9月10日(1999.9.10)  
(31) 優先権主張番号 6 0 / 0 7 6 , 6 6 7  
(32) 優先日 平成10年3月3日(1998.3.3)  
(33) 優先権主張国 米国 (U S)

(71) 出願人 シーベル システムズ, インコーポレイテ  
ィド  
アメリカ合衆国, カリフォルニア 94402,  
サン マテオ, サウス グラント ストリ  
ート 1855  
(72) 発明者 アルナイス, ヘルベルト  
アメリカ合衆国, カリフォルニア 94065,  
レッドウッド シティ, ショール サーク  
ル 529  
(74) 代理人 弁理士 石田 敬 (外4名)

最終頁に続く

(54) 【発明の名称】 ソフトウェア・アップグレードの分配およびインスタンス化のための方法、システム、装置およびプログラム製品

(57) 【要約】

ソフトウェア・バージョン・アップグレードは、分散計算環境において分配され、インスタンス化される。これには、ソフトウェア・バージョン・アップグレード・キットの内容としてソフトウェアの要求されるバージョンを定義することが含まれる。ソフトウェア・アップグレード・キットは、例えばテーブル・オブ・コンテンツまたはリストとしてデータベースに書き込まれ、ソフトウェア・アップグレード・キット・テーブルを生成する。このテーブルは、ソフトウェア・バージョン・アップグレード・キットを生成するために使用される。キットは、次に、アップグレードされるクライアントにアップグレード・キットのコピーをダウンロードするためのソースとして使用される。これにより、クライアントのソフトウェアをアップグレードすることができる。

**【特許請求の範囲】**

【請求項1】 クライアント・サーバー・システムにおけるクライアント・コンピュータ上にソフトウェア・アップグレードを分配し及びインストールする方法であって、

クライアントに関してインストールされたソフトウェアの最小バージョン及び最大バージョンを設定することと、

該インストールされたソフトウェアの最小バージョンおよび最大バージョンから、1つ又はそれ以上のソフトウェア・コンポーネントを1つのバージョンから別のバージョンにアップグレードするためのファイル、動作およびアップグレード・ウィザードからなるソフトウェア・バージョン・アップグレード・キットの内容を定義することと、

アップグレード・キットの内容及び該当のアップグレードがどのソフトウェア・コンポーネントを必要とするかを示すテーブル・オブ・コンテンツとしてソフトウェア・バージョン・アップグレード・キットの内容をデータベースに書き込むことと、

クライアント上のインストールされたソフトウェアに質問して該クライアント上の該インストールされたバージョンを判定することと、

ソフトウェア・バージョン・アップグレード・キットのテーブル・オブ・コンテンツをクライアント上にインストールされたソフトウェアと比較して、現在インストールされたソフトウェア・バージョンから要求されるソフトウェア・バージョンにアップグレードするためにクライアント上に呼び出される必要のあるソフトウェア・バージョン・アップグレード・キットを判定することと、

該テーブル・オブ・コンテンツから該ソフトウェア・アップグレード・キットを構成することと、

該サーバーからアップグレードされるべきクライアントに該ソフトウェア・アップグレード・キットのコピーをダウンロードすることと、

該アップグレードを適用する前にローカル・ファイルのバックアップを作成することと、

ソフトウェア・バージョン・アップグレードのために実行されるべきアップグ

レード・キット・アイテムのリストをドライバ・ファイルから読み取るとともに実行されるべきアップグレードのリスト及び各アップグレードの状態を状態ファイルから読み取るアップグレード・ウィザードを、クライアント上のソフトウェアをアップグレードするために呼び出すことと、

を具備する方法。

【請求項2】 アップグレードの進行を追跡し、エラーが生じた場合変更をロールバックし、バックアップされたローカル・ファイルをインスタンス化し、及びセーブ・ポイントからアップグレードをリスタートする、請求項1に記載の方法。

【請求項3】 サーバーからクライアント上のソフトウェア・アップグレードの進行を監視することを具備する、請求項1に記載の方法。

【請求項4】 クライアントがアップグレードを要求する前に、1つ又はそれ以上のアップグレード・キットをクライアントにダウンロードする、請求項1に記載の方法。

【請求項5】 アップグレードされるソフトウェアがデータベース管理ソフトウェアであり、アップグレードがさらにデータベース・スキーマ変更を含む、請求項1に記載の方法。

【請求項6】 サーバー・コンピュータ及び該サーバー・コンピュータと周期的にドッキングする少なくとも1つのクライアント・コンピュータを有するクライアント・サーバー・コンピュータ・システムであって、前記サーバー・コンピュータがデータベース管理システム及びその上のデータベースを有し、前記データベースはドッキング中にサーバー・コンピュータからクライアント・コンピュータに部分的に複製され、前記サーバー及び前記クライアントは更にサーバー・コンピュータからクライアント・コンピュータ上に常駐するソフトウェアにアップグレードを適用するためのアップグレード・プログラム・コードを有し、前記アップグレード・プログラム・コードが、

クライアント・コンピュータに関してインストールされたソフトウェアの最小バージョン及び最大バージョンを設定するプログラム・コードと、

該インストールされたソフトウェアの最小バージョンおよび最大バージョンが

ら、1つ又はそれ以上のソフトウェア・コンポーネントを1つのバージョンから別のバージョンにアップグレードするためのファイル、動作およびアップグレード・ウィザードからなるソフトウェア・バージョン・アップグレード・キットの内容を定義するプログラム・コードと、

アップグレード・キットの内容及び該当のアップグレードがどのソフトウェア・コンポーネントを必要とするかを示すテーブル・オブ・コンテンツとしてソフトウェア・バージョン・アップグレード・キットの内容をデータベースに書き込むプログラム・コードと、

クライアント上のインストールされたソフトウェアに質問して該クライアント上の該インストールされたバージョンを判定するプログラム・コードと、

ソフトウェア・バージョン・アップグレード・キットのテーブル・オブ・コンテンツをクライアント上にインストールされたソフトウェアと比較して、現在インストールされたソフトウェア・バージョンから要求されるソフトウェア・バージョンにアップグレードするためにクライアント上に呼び出される必要のあるソフトウェア・バージョン・アップグレード・キットを判定するプログラム・コードと、

該テーブル・オブ・コンテンツから該ソフトウェア・アップグレード・キットを構成する該サーバー上のプログラム・コードと、

該サーバーからアップグレードされるべきクライアントに該ソフトウェア・アップグレード・キットのコピーをダウンロードするプログラム・コードと、

該アップグレードを適用する前にローカル・ファイルのバックアップを該クライアント・コンピュータ上に作成するプログラム・コードと、

ソフトウェア・バージョン・アップグレードのために実行されるべきアップグレード・キット・アイテムのリストをドライバ・ファイルから読み取るとともに実行されるべきアップグレードのリスト及び各アップグレードの状態を状態ファイルから読み取るアップグレード・ウィザードを、クライアント上のソフトウェアをアップグレードするために呼び出すプログラム・コードと、

を具備するクライアント・サーバー・コンピュータ・システム。

【請求項7】 アップグレードの進行を追跡し、エラーが生じた場合変更を

ロールバックし、バックアップされたローカル・ファイルをインスタンス化し、及びセーブ・ポイントからアップグレードをリスタートするためのプログラム・コードを更に含む、請求項6に記載のクライアント・サーバー・コンピュータ・システム。

【請求項8】 サーバーからクライアント上のソフトウェア・アップグレードの進行を監視するためのプログラム・コードを更に含む、請求項6に記載のクライアント・サーバー・コンピュータ・システム。

【請求項9】 ドッキング中クライアントがアップグレードを要求する前に1つ又はそれ以上のアップグレード・キットをクライアントにダウンロードしドッキング後に該アップグレードをインストールするためのプログラム・コードを更に含む、請求項6に記載のクライアント・サーバー・コンピュータ・システム。

【請求項10】 サーバー・コンピュータからクライアント・コンピュータ上に常駐するソフトウェアにアップグレードを適用するためのアップグレード・プログラム・コードが、該クライアント・コンピュータに常駐するデータベース・スキーマをアップグレードするためのプログラム・コードを更に含む、請求項6に記載のクライアント・サーバー・コンピュータ・システム。

【請求項11】 サーバー・コンピュータからクライアント・コンピュータ上に常駐するソフトウェアにソフトウェア・アップグレードを適用するクライアント・サーバー・コンピュータ・システムにおいて使用されるプログラム製品であって、前記プログラム製品は該サーバー・コンピュータ及び該クライアント・コンピュータ上の記憶媒体に存在するプログラム・コードをアップグレードするソフトウェアからなり、前記ソフトウェア・アップグレード・プログラム・コードが、

クライアント・コンピュータに関してインストールされたソフトウェアの最小バージョン及び最大バージョンを設定するプログラム・コードと、

該インストールされたソフトウェアの最小バージョンおよび最大バージョンから、1つ又はそれ以上のソフトウェア・コンポーネントを1つのバージョンから別のバージョンにアップグレードするためのファイル、動作およびアップグレー

ド・ウィザードからなるソフトウェア・バージョン・アップグレード・キットの内容を定義するプログラム・コードと、

アップグレード・キットの内容及び該当のアップグレードがどのソフトウェア・コンポーネントを必要とするかを示すテーブル・オブ・コンテンツとしてソフトウェア・バージョン・アップグレード・キットの内容をデータベースに書き込むプログラム・コードと、

クライアント上のインストールされたソフトウェアに質問して該クライアント上の該インストールされたバージョンを判定するプログラム・コードと、

ソフトウェア・バージョン・アップグレード・キットのテーブル・オブ・コンテンツをクライアント上にインストールされたソフトウェアと比較して、現在インストールされたソフトウェア・バージョンから要求されるソフトウェア・バージョンにアップグレードするためにクライアント上に呼び出される必要のあるソフトウェア・バージョン・アップグレード・キットを判定するプログラム・コードと、

該テーブル・オブ・コンテンツから該ソフトウェア・アップグレード・キットを構成する該サーバー上のプログラム・コードと、

該サーバーからアップグレードされるべきクライアントに該ソフトウェア・アップグレード・キットのコピーをダウンロードするプログラム・コードと、

該アップグレードを適用する前にローカル・ファイルのバックアップを該クライアント・コンピュータ上に作成するプログラム・コードと、

ソフトウェア・バージョン・アップグレードのために実行されるべきアップグレード・キット・アイテムのリストをドライバ・ファイルから読み取るとともに実行されるべきアップグレードのリスト及び各アップグレードの状態を状態ファイルから読み取るアップグレード・ウィザードを、クライアント上のソフトウェアをアップグレードするために呼び出すプログラム・コードと、

を具備するプログラム製品。

【請求項12】 アップグレードの進行を追跡し、エラーが生じた場合変更をロールバックし、バックアップされたローカル・ファイルをインスタンス化し、及びセーブ・ポイントからアップグレードをリスタートするためのプログラム

・コードを更に含む、請求項11に記載のプログラム製品。

【請求項13】 サーバーからクライアント上のソフトウェア・アップグレードの進行を監視するためのプログラム・コードを更に含む、請求項11に記載のプログラム製品。

【請求項14】 ドッキング中クライアントがアップグレードを要求する前に1つ又はそれ以上のアップグレード・キットをクライアントにダウンロードしドッキング後に該アップグレードをインストールするためのプログラム・コードを更に含む、請求項11に記載のプログラム製品。

【請求項15】 サーバー・コンピュータからクライアント・コンピュータ上に常駐するソフトウェアにアップグレードを適用するためのアップグレード・プログラム・コードが、該クライアント・コンピュータに常駐するデータベース・スキーマをアップグレードするためのプログラム・コードを更に含む、請求項11に記載のプログラム製品。

**【発明の詳細な説明】****【0001】****発明の分野**

本発明は、ソフトウェア更新をオンラインで提供するためのシステム、方法およびプログラム製品に関するものである。更新は、一連の個別のコンピュータのうち1台またはそれ以上にまたはコンピュータのネットワークに提供することができる。望ましい実施例においては、ソフトウェア・更新は、クライアント・サーバー・ネットワーク、たとえば部分的に複製される関係データベースまたはトランザクション処理システムを実行するタイプのクライアント・サーバー・ネットワークの個々のクライアント、WAN またはLAN 上の企業に提供される。

**【0002】****発明の背景**

企業は、その業務を行うために多数の端末、すなわちデスクトップ・コンピュータ、ポータブル・コンピュータおよびネットワーク端末を利用する。これらの端末全てが同一のイメージ、インターフェイスおよびソフトウェアを持つことが望ましいが、1つの企業の中にある範囲の年代および機能の端末が存在する場合、これは必ずしも可能ではない。しかし、1つの企業の中の端末が効果的に通信できることが絶対に必要である。このためには、ソフトウェア・マイグレーションを慎重に計画し調整する必要がある。

**【0003】**

1つまたはそれ以上の中央データベース（対応データベースおよびLotus Notes、Novell GroupWise などのグループウェアにおいて想定されるデータベースを含む）およびローカルに部分的に複製されるデータベースを含むシステムにおいては、しばしば特定の企業のニーズ、また時には企業の部局内の特定のユーザーおよびユーザーのグループのニーズに合わせてカスタマイズされたソフトウェアが大量にある。ソフトウェアを構成し、カスタム・モジュールおよびオブジェクトを書き込むには膨大な作業を必要とする。ソフトウェアが新しいリリースにアップグレードされる場合、新しいリリースを構成し以前のバージョンのカスタマイズの機能を実現し直しながら、整合的にアップグレードするには、かなりのプ



ログラミング時間が必要である。

【0004】

従って、企業が構成およびオブジェクトなどソフトウェアの1つのバージョンから別のバージョンへの変更を速やかに移行できるようにする機能を提供することが望ましい。

ソフトウェア・アップグレードのオンライン分配を使用することによりユーザー・フレンドリーなオンラインの方法でソフトウェア・アップグレードを速やかに移行することが望ましい。

【0005】

発明の目的

以下のビジネス上の要件を満たしながら、ソフトウェア・アップグレードをオンライン分配することが本発明の目的である。

本発明の主な目的は、データベース・スキーマを同期化することである。

さらに本発明の目的は、企業内のあらゆるタイプのインストレーションおよび端末をサポートすることである。すなわち、ソフトウェア・プロバイダまたはアップグレードを行う者例えばデータベース管理者は、接続されるクライアント、リモート・クライアント、地域サーバーおよびアプリケーション・サーバーを含めて企業内のあらゆるタイプのインストレーションをアップグレードできなければならない。

【0006】

さらに本発明の目的は、第三者ソフトウェアのアップグレードをサポートすることである。

さらに本発明の目的は、企業内にインストールされるソフトウェアの全面的および部分的アップグレードをサポートすることである。

付加的目的はリスタート可能性である。これは、アップグレード・ソフトウェアが、アップグレードの進行を追跡して、エラーが生じた場合には自動的に変更をロールバックしなければならない、また最終セーブ・ポイントからアップグレードをリスタートすることができなければならないことを意味する。

【0007】

特に異種混合および（または）地理的分散を特徴とする企業においては、本発明の重要な目的は、集中定義および監視である。アップグレードを行う者例えばデータベース管理者は、中央ロケーションからソフトウェアおよびスキーマ・アップグレードを簡単に定義し分配して、インストレーション全てのアップグレードの進行を監視できなければならない。

#### 【0008】

アップグレードは自立できなければならない。つまり、アップグレードは、他のプログラムまたはデータに依存せずに、独自に実行できなければならない。

最後に、以上の目的は、使いやすさおよびユーザー・フレンドリネスを維持しながら達成され、守られなければならない。

#### 発明の要約

以上の目的およびその他の目的が、本発明の方法、装置、システムおよびプログラム製品によって達成される。

#### 【0009】

本発明は、ソフトウェア・バージョン・アップグレードおよびデータベース・スキーマ同期化を提供する。ソフトウェア・プロバイダまたはアップグレードを行う者例えばデータベース管理者は、移動データベースおよび地域データベースを含めてあらゆるリモート・データベースに対してデータ・ベース・スキーマ変更を適用し、分散する。さらに、ソフトウェア・プロバイダまたはアップグレードを行う者例えばデータベース管理者は、微細なスキーマ変更（例えば、データ・ベース拡張）、パッチおよび大幅なスキーマ変更を手操作介入なしに適用する。

#### 【0010】

本発明に従えば、あらゆるタイプのインストレーションにソフトウェア・バージョンアップグレードを提供できる。すなわち、ソフトウェア・プロバイダまたはアップグレードを行う者例えばデータベース管理者は、接続されるクライアント、リモート・クライアント、地域サーバーおよびアプリケーション・サーバーを含めてあらゆるタイプのインストレーションおよび端末をアップグレードすることができる。

## 【0011】

本発明の1つの態様は、分散計算環境においてソフトウェア・バージョン・アップグレードを分配しインスタンス化する方法である。この方法は、インストールされるソフトウェアのバージョンの最小(および最大)レベルを設定すること、およびこのレベルを使用してソフトウェア・バージョン・アップグレード・キットの内容を定義することを含む。ソフトウェア・バージョン・アップグレード・キットの内容は、例えばテーブル・オブ・コンテンツなどとしてデータベースに書き込まれ、ソフトウェア・バージョン・アップグレード・キット・テーブルを生成するために使用される。このテーブルは、ソフトウェア・バージョン・アップグレード・キットを構成するために使用される。アップグレード・キットのコピーはアップグレードされるクライアントにダウンロードされ、クライアントのソフトウェアがアップグレードされる。

## 【0012】

ソフトウェア・アップグレード・キットのテーブル・オブ・コンテンツはインストールされた後、クライアントにローカルにインストールされているソフトウェアと比較され、クライアントにインストールする必要のあるソフトウェアがクライアントにダウンロードされる。この比較は、サーバー、クライアントまたは特定のソフトウェア・コンポーネントのスタートアップ時に行うことができる。

## 【0013】

クライアントが移動クライアントである場合、比較およびダウンロードはドッキング・セッションに行われ、現ドッキング・セッションまたはその後のドッキング・セッション中にアップグレードがインスタンス化される。

本発明の1つの実施例においては、分散計算環境は、複製されたまたは部分的に複製されたデータベースを持つデータベース管理システムであり、クライアントのデータベース・スキーマ・バージョンがサーバーのデータベース・スキーマ・バージョンと比較されて、クライアントとサーバーのデータベース・スキーマ・バージョンが一致しない場合にはクライアントのデータベース・バージョンが再初期化され、クライアントとサーバーのデータベース・スキーマ・バージョンが一致する場合にはクライアントのデータベースがサーバーのデータベースと同

期化される。

【0014】

本発明によれば、第三者ソフトウェアがサポートされる。ソフトウェア・プロバイダまたはアップグレードを行う者例えばデータベース管理者は、カスタマ・インストレーションが依存する第三者ソフトウェアをアップグレードすることができる。例えば、ソフトウェア・プロバイダ、管理情報サービス・スーパーバイザまたはアップグレードを行う者例えばデータベース管理者は、カスタマ・インストレーションに常駐するMicrosoft Word、Adobe Acrobat、Microsoft Accessなどをアップグレードすることができる。

【0015】

本発明の方法および装置に従って全面的および部分的アップグレードが可能である。アップグレードを行う者例えばデータベース管理者は、インストレーションの全部または一部をアップグレードできる。例えば、アップグレードを行う者例えばデータベース管理者は、完全に新しいインストレーションを分配することもできるし、CDF ファイル、実行可能ファイル、DLL、レポートなど既存のインストレーションのパッチを分配することもできる。同様に、アップグレードを行う者例えばデータベース管理者は、様々な言語のアップグレードを分配することができる。

【0016】

上記の目的は、使いやすさおよびユーザー・フレンドリネスを維持しながら達成される。この点に関して、使いやすさのためには、既存のプログラムの自動検出およびクライアントおよび複製エージェントを含めてアップグレード・プログラムの自動呼出しが必要である。インストール・ソフトウェアは、ユーザーが明示的にアップグレードを探し選択する必要があるように、アップグレードを利用できローカル・インストレーションに応用できるときを自動的に検出しなければならない。ユーザー・フレンドリネスのためには、ソフトウェア・アップグレード・プログラムはユーザーがアップグレードをキャンセルまたは延期できるようにしなければならない。ソフトウェア・アップグレード・プログラムは、アップグレードの状態についてユーザーに常に情報を与え、未決のアップグレード、ア

アップグレードの際にユーザーが実施すべきこと、アップグレードに必要な推定時間およびアップグレード中の進行指標をユーザーに知らせなければならない。この中には自己アップグレードが含まれ、アップグレードは自身をアップグレードできなければならない、アップグレードを行う者たとえばデータベース管理者は、アップグレード・プログラムのアップグレードを分配できなければならない。ユーザー・フレンドリネスには、適応性も含まれる。すなわち、アップグレード・ドウィザードを使用するか否かに関係なく個々のユーザーがソフトウェア・コンポーネントをインストールまたは更新するための方法を使用するために、ソフトウェア・コンポーネントの現状態を使用してアップグレードが必要かどうか識別する。

#### 【0017】

これに関連する要件はリスタート可能性である。すなわち、アップグレード・ソフトウェアは、アップグレードの進行を追跡して、エラーが生じた場合には変更を自動的にロールバックしなければならない、アップグレード・ソフトウェアは最後のセーブ・ポイントからアップグレードをリスタートできなければならない。

#### 【0018】

望ましい特性は、早期ダウンロードの可能性である。これにより、アップグレードが必要とされる前でも、移動ユーザーはアップグレード・キットをそのローカル・マシンにダウンロードして、応用することができる。これにより移動ユーザーは、時間があるときにまたはファスト・ネットワークにアクセスできるとき（例えばHQ訪問）にアップグレードをダウンロードできる。

#### 【0019】

さらに望ましい特性は、集中定義および監視であり、アップグレードを行う者例えばデータベース管理者は、中央ロケーションからソフトウェアおよびスキーマ・アップグレードを簡単に定義し分配できなければならない。さらに、アップグレードを行う者例えばデータベース管理者は、インストレーション全てのアップグレードの進行を監視できなければならない。例えば、アップグレードを行う者例えばデータベース管理者は、アップグレードが応用された全てのインストレ

ーションを見つけられなければならない。これには、保証つきデリバリーが含まれる。全てのアップグレードは全てのインストレーションに分配され応用される。アップグレードを行う者例えばデータベース管理は、各インストレーションのアップグレード状態を明示的に監視する必要があるようにしなければならない。

#### 【0020】

本発明は、テストおよび生産環境をサポートする。アップグレードを行う者例えばデータベース管理者は、テスト環境においてアップグレードを構成ししテストできなければならない、その後全ての生産ユーザーに分配するために生産環境に変更を簡単に移行できなければならない。

本発明により応用されるアップグレードは自立できる。すなわち、アップグレードは他のプログラムまたはデータに依存しないで独自に実行できる。

#### 【0021】

本発明に従ったアップグレードのインストールは、ネットワーク・トラフィックをわずかしかな必要としない。アップグレードを必要とするサイトだけにこれを送信することによりファイル転送を最小限に抑えることが重要である。例えば、アプリケーション・サーバー・アップグレードは、地域データ・インストレーションにしか送信されない。Microsoft Wordアップグレードはクライアント・インストレーションにしか送信されない。

#### 【0022】

本発明は、付属の図面および表を参照することによって、より良く理解できる。

### 本発明の詳細な説明

#### インプリメンテーションの概要

アップグレードの定義、アップグレードの開始および分配およびエンドユーザー・レベルのアップグレードについては図に示されるとおりである。

#### 【0023】

図1 に示されるとおり、クライアント (101 ) として示される管理者は、アップグレード・キットを定義する。クライアント (101 ) は、このアップグレード・キット定義をデータベース・サーバーに常駐するサーバー・データベース (

103) に書き込み、サーバー・データベース(103)においてアップグレード・キット・テーブル(104)を生成する。クライアント(101)においてアップグレードを定義するステップはアップグレード・キット・ファイル(102)も生成する。クライアントは、このアップグレード・キット・ファイル(102)を、たとえば後のダウンロードのためにファイル・サーバー(105)にコピーする。アップグレード・キット・ファイル(106)がリリースされるとき、アップグレード・キット・アーカイブ・ファイルが生成される。

#### 【0024】

図2は、アップグレードの開始および分配を示している。図2に示されるプロセスにおいて、管理者(101)は、ソフトウェア・アイテムについて要求されるバージョンを更新する。

これにより、常駐ソフトウェア・コンポーネント・テーブル(104A)を持つデータベース・サーバー(103)は、スタートアップされると、ソフトウェア要求されるのバージョンを接続される各ローカル・ユーザー(211)ごとに各ソフトウェア・アイテムごとにローカルにインストールされているバージョンと比較する。アップグレードが必要とされる場合、ローカル・ユーザー(211)はファイル・サーバー(105)からアップグレード・キット・アーカイブ・ファイル(106)を検索し、ローカルにアップグレードを応用するためにアップグレード・ウィザードを呼び出す。

#### 【0025】

図2には、移動ユーザー(215)のアップグレードも示されている。移動ユーザー(215)の場合、アプリケーション・サーバー(213)が、要求される新しいバージョンを移動ユーザー(215)および地域ユーザーに送る。ドッキング・セッション後、ドッキング・クライアントは、要求されるバージョンをローカルにインストールされているバージョンと比較する。アップグレードが必要とされる場合、ドッキング・クライアントはファイル・サーバー(105)からアップグレード・キット・アーカイブ・ファイル(106)を検索して、ローカル・マシン(215)にアップグレードを応用するためにアップグレード・ウィザードを呼び出す。

## 【0026】

図3 は、カスタマのための一連のテストおよび分配手順を示している。図3 に示されるとおり、主レポジトリ (305 ) においてアップグレードCDまたはディスク (307 ) が作成され、サーバー・テスト・ユーザー (313 ) 、接続されるテスト・ユーザー (311 ) および移動テスト・ユーザー (315 ) を含むテスト・ユーザーに分配するためにテスト・データベース・サーバー (303 ) に出荷される。テスト・ユーザー (311 、 313 および315 ) はそのテスト環境をアップグレードし、テストのためにソフトウェアのアップグレード・バージョンを分配する。カスタマ (311 、 313 、 315 ) はソフトウェアをテストする。テストが合格であれば、ソフトウェアの新たにアップグレードされたバージョンが生産ユーザー (4121、 413 、 415 ) に分配される。

## 【0027】

図4 は、リモート・アップグレードのためのデータ・モデルを示している。この図は、アップグレード・キット (1001) 、コンポーネント (1005) および関連コンポーネント (1007) を含むキット・コンポーネント (1003) 、キット・アイテム (1009) およびキット・アイテム引数 (1011) 、およびキット・アイテム・タイプ (1013) を示す。

## ソフトウェア・コンポーネント

アップグレードの基本ユニットはソフトウェア・コンポーネントである。ソフトウェア・コンポーネントは、アップグレードによって個別にインストールし、追跡することができる1組のソフトウェアである。各ソフトウェア・コンポーネントは、ある範囲の要求されるバージョンを持っている。ソフトウェア・ベンダ、MIS スーパーバイザまたはアップグレードを行う者例えばデータベース管理者は、各ソフトウェア・コンポーネントごとに要求されるバージョンの範囲を指定する。単なる例として(制限的でなく)示すと、Siebel Remote アプリケーションにおけるソフトウェア・コンポーネントの例は、Siebel Schema 、 Siebel Client 、 Siebel Server およびMicrosoft Wordである。

## アップグレード・キット

ソフトウェア・ベンダ、MIS スーパーバイザまたはアップグレードを行う者例



例えばデータベース管理者は、1 つまたはそれ以上のソフトウェア・コンポーネントをインストールまたはアップグレードするためにアップグレード・キットを作成する。アップグレード・キットは、1 つまたはそれ以上のソフトウェア・コンポーネントをあるバージョンから別の(通常高位の)バージョンにアップグレードするための一群のファイルおよび動作である。ソフトウェア・コンポーネントのアップグレードを開始するために、例えば、アップグレードを行う者例えばデータベース管理者は、ソフトウェア・コンポーネントの新しいバージョンをインストールするためにアップグレード・キットを構成し、リリースする。その後、アップグレードを行う者例えばデータベース管理者は、ユーザーが確実にソフトウェア・コンポーネントの要求される新しいバージョンにアップグレードするように、ソフトウェア・コンポーネントの要求されるバージョンを更新する。

#### 自動アップグレード検出

予め定められた時点で(例えば、クライアントのスタートアップ、各同期化セッションの前後、第三者ソフトウェア・コンポーネントの呼び出し前、サーバーのスタートアップ時)、システム・プログラムは、現在実行中のプログラムについてソフトウェア・コンポーネントが最新のものであるかどうかを確認する。例えば、Siebel Remote の場合、Siebel Client はSiebel SchemaおよびSiebel Client コンポーネントのバージョンをチェックするが、必ずしもSiebel Server コンポーネントをチェックしない。サーバーは、ソフトウェア・コンポーネント自体に応答指令を發し、インストールされているバージョンと要求されるバージョンを比較することによって、ソフトウェア・コンポーネントの現在インストールされているバージョンを判定する。

#### 【0028】

あるソフトウェア・コンポーネントが要求されるバージョンの範囲内にない場合、サーバーは全てのソフトウェア・コンポーネントのバージョンをチェックし、1 つまたはそれ以上のアップグレード・キットを探し、古くなったソフトウェア・コンポーネントを要求されるバージョンにアップグレードする。その後、サーバーは、要求されるアップグレードについてユーザーに知らせ、アップグレードを実行するようユーザーにプロンプトを示す。

### アップグレード・ウィザード

サーバーは、アップグレード・キットを応用するためにアップグレード・ウィザードを呼び出す。アップグレード・ウィザードを呼び出した後、プログラムは、アップグレードする必要のあるデータベース・テーブルまたはファイルのロックを解除するように、終了(エグジット)する。例えば、クライアントは、アップグレード・ウィザードを呼び出して、終了する。サーバーは、現在アクティブの全てのサーバー・コンポーネントにシャットダウン・メッセージを送り、全てのサーバー・コンポーネントが停止するのを待ち、アップグレード・ウィザードを呼び出して、終了する。

#### 【0029】

アップグレード・ウィザードは、アップグレード・キットをローカル・マシンに読み込み応用することのできる、単一のスタンドアローン実行可能ファイル(DLLを必要としない)であることが望ましい。アップグレード・ウィザードは、アップグレードの進行を追跡し、自動的にエラーから回復する。例えば、アップグレード・ウィザードは、アップグレード・キットを応用しながら、ローカル・データベースおよびローカル・ファイルのバックアップを作成する。アップグレード中エラーが生じると、アップグレード・ウィザードは、全ての変更をロールバックしようとし、ローカル・マシンを原状態に復元する。

#### 【0030】

クライアントまたはサーバーがスタートすると、アップグレードが進行中かどうかまたは失敗したかどうかを検出する。アップグレードが失敗した場合、ユーザーにこれを知らせて、アップグレードをリスタートするためにアップグレード・ウィザードを呼び出す。クライアントの場合、ユーザーがアップグレードのリスタートを辞退する場合、クライアントは、読取専用モードを続ける。読取り専用モードにおいては、クライアントは、ユーザーがデータを見ることができるようにするが、ユーザーがデータを修正することを防ぐ。サーバーの場合、アップグレードを行う者例えばデータベース管理者がアップグレードのリスタートを辞退する場合、サーバーは、エラー・メッセージを発して、停止する。

#### 【0031】

アップグレードが順調に完了した後、ローカル・マシンのソフトウェア・コンポーネントは最新のものであるはずである。アップグレード・ウィザードは、アップグレードを呼び出したオリジナル・プログラムをリスタートさせる。

#### アップグレードの使用

この節では、アップグレードの定義、分配および応用のプロセスをさらに詳しく説明する。

#### カスタマイズ

アップグレードを行う者例えばデータベース管理者は、その環境に合わせてデータベース管理システムをカスタマイズするために、ツール・プログラムおよびその他のプログラムを使用する。アップグレードを行う者例えばデータベース管理者は、データベース・スキーマを拡張し、クライアント画面、リポート・ファイルおよびその他のファイルをカスタマイズする能力を持っている。

#### アップグレードの定義

アップグレードを行う者例えばデータベース管理者は、サーバー・データベースにおけるアップグレードを定義する。このプロセスは、ソフトウェア・コンポーネント、および各ソフトウェア・コンポーネントを特定のバージョンにアップグレードするために必要とされるアップグレード・キットを定義することから成る。

#### ソフトウェア・コンポーネントの定義

アップグレードを行う者例えばデータベース管理者は、新しいソフトウェア・コンポーネントを定義するためにソフトウェア・コンポーネント画面またはツールを使用する。ソフトウェア・コンポーネントは、アップグレードによって個別にインストールし、追跡し、デインストールできる1組のソフトウェアである。本文書において使用される場合、ソフトウェア・コンポーネントは以下の属性を持つ：

名前： ソフトウェア・コンポーネントを識別するための名前。

#### 【0032】

要求されるバージョン： 各マシンにインストールされなければならないバージョンの範囲すなわち最小および最大バージョン。

ロケート方法： ソフトウェア・コンポーネントがあるマシンにインストールされているかどうかを検出するための方法の指定。例えば、Microsoft Windows がローカル・マシンのどこにインストールされているかを識別するためにWindows レジストリを読み取る。

【0033】

バージョン方法： マシンにインストールされているソフトウェア・コンポーネントの現バージョンを識別するための方法の指定。例えば、ローカル・マシンに現在インストールされているMicrosoft Words のバージョンを得るためにWindows レジストリを読み取る。

「コンパイル済み」のコンポーネント情報： ソフトウェア・コンポーネントの以前にリリースされたバージョンに関する情報。例えば、Siebel Remote は、2つのバージョンのソフトウェア・コンポーネントを記憶する。ユーザーは、現バージョンを見て操作することができる。Siebel Remote は、ソフトウェア・コンポーネント・テーブル（すなわちS \_\_UPG \_\_COMP）のLONGカラムに以前にリリースされたバージョンを記憶する。アップグレードを行う者例えばデータベース管理者は「コンパイル済み」のLONGカラムに現バージョンをコピーするためにコンポーネント画面を使用する。

【0034】

スキーマ： スキーマは、メタデータベースとも呼ばれるデータベース・スキーマである。これには、実行するためにデータベースにインストールしなければならないデータベース・テーブル、インデクス、ビュー、シード・データおよびレポジトリ・データが含まれる。

クライアント： クライアントを実行するために必要な実行可能ファイル、dll、リポート、ヘルプ・ファイル。

【0035】

CDF： クライアントに関するカスタマの構成の定義。

サーバー： サーバーを実行するために必要な実行可能ファイル、dll およびその他のファイル。

アップグレード・ウィザード： アップグレード・キットをマシンに応用する

スタンドアローン実行可能ファイル。

【0036】

ほとんどのユーザーにとって、上に説明したソフトウェア・コンポーネントの定義済みセットで完璧である。必要な場合には、アップグレードを行う者例えばデータベース管理者は、ユーザーに分配し応用したいソフト系・コンポーネントを定義することもできる。

アップグレード・キット画面を使用するアップグレード・キットの定義

本発明の望ましい実施態様において、アップグレードを行う者例えばデータベース管理者は、1 つまたはそれ以上のソフトウェア・コンポーネントをインストールするために、アップグレード・キット画面を使用してアップグレード・キットを定義する。

【0037】

各アップグレード・キットは、ソフトウェア・コンポーネントをインストールするために必要な全てのファイルおよびコマンドを含まなければならない。例えば、アップグレード・キットを使用して、全てのクライアントにMicrosoft Word 7.0.2をインストールすることができる。アップグレード・キットは、Microsoft Word 7.0.2をインストールするための全てのファイル（例えば、実行可能ファイル、サンプル・ファイル、テンプレートなど）およびコマンド（例えば、レジストリの更新、ショートカットの構成など）を含んでいなければ成らない。アップグレードを行う者例えばデータベース管理者は、アップグレード・キットを定義する際以下の情報を提供する：

アップグレード・キット情報： アップグレード・キットの名前、タイトルおよび説明を示す。

【0038】

アップグレード・キット・アイテム： アップグレードを行う者例えばデータベース管理者は、アップグレード・キットを応用するためにアップグレード・ウィザードが順次実行する1組の動作（またはアイテム）を定義する。Error! Reference source not found は、可能性のあるアップグレード・キット・アイテムのリストを示す。アップグレードを行う者例えばデータベース管理者は

、アップグレード・キットのためのファイルを予め作成し、これをアップグレード・キットと一緒に記録する。

**【0039】**

要求されるコンポーネント・バージョン： アップグレードを行う者例えばデータベース管理者は、アップグレード・キットを応用するためにインストールされなければならない要求されるソフトウェア・コンポーネントのリストも定義する。例えば、データをデータベースにインポートするためのアップグレード・キットを応用するためには、SQLAnywhere v5.5をインストールする必要があるかも知れない。

**【0040】**

インストールされるコンポーネント・バージョン： アップグレードを行う者例えばデータベース管理者は、アップグレード・キットが応用されるソフトウェア・コンポーネントのリストも定義する。例えば、あるファイルを実行するためのアップグレード・キットは、Microsoft Word 7.0.2をインストールするかも知れない。

**【0041】**

アップグレード・キット画面は、カスタマ・サイトにおけるアップグレード・キットの作成を助けるためにウィザードを示す。アップグレード・キット・ウィザードは、アップグレードを行う者例えばデータベース管理者が、ユーザーがカスタマイズできるソフトウェア・コンポーネントのためのアップグレード・キットを作成できるようにする。

**【0042】**

アップグレードを行う者例えばデータベース管理者は、アップグレードの定義を終了した後、例えばSiebelアップグレード・キット画面上の「リリース」ボタンを押してコマンドを発することによって結果をセーブして、全てのインストレーションがアップグレード・キットを利用できるようにする。コマンド例えばボタンは、以下のことを行う：

アップグレード・キットのためのアップグレード・ウィザード・ドライバ・ファイルを生成する。

**【0043】**

アップグレード・キット・アーカイブ・ファイルを作成し、これをファイル・システムに記憶する。アーカイブ・ファイルは、アップグレード・ウィザード・ドライバ・ファイルおよびマシンにアップグレード・キットを応用するために必要な全てのファイルから成る。

アップグレード・キット情報を「コンパイル済みの」LONG カラムにコンパイルする。この「コンパイル済み」のカラムは、プログラムがアップグレード・キットを探すために必要な全ての情報を記憶する。この「コンパイル済みの」カラムは、アップグレード・キットを探すための時間を短縮するために設けられる。

**アップグレードの配備**

アップグレードの定義後、アップグレードを行う者例えばデータベース管理者は、サーバー・データベースからアップグレードを開始する。全てのロケーションのクライアント・プログラムは、自動的にアップグレードを検出し必要に応じてこれを応用する。

**アップグレード・キットの早期ダウンロード**

アップグレード・キットが定義された後、移動データベースおよび地域データベースにアップグレード・キット定義関連ファイルを分配することができる。この場合、移動ユーザーは、アップグレード・キット画面にナビゲートし、次の同期化セッションにデータベース管理システムがアップグレード・キット・アーカイブ・ファイルをダウンロードするよう要求できる。これにより、移動ユーザーは、ローカル・マシンにアップグレード・キットを応用する必要が生じるずっと前に大量のアップグレード・キット・アーカイブ・ファイルをダウンロードできる。例えば、移動ユーザーは本部を訪問して、モデムを通じて行うよりずっと短い時間でLAN を通じてアップグレード・キット・アーカイブ・ファイルをダウンロードできる。

**【0044】**

アップグレード・キットの早期ダウンロードは移動ユーザーにとって強制的なものではない。移動ユーザーは、アップグレード・キット・アーカイブ・ファイ

ルを明示的に要求する必要はない。データベース管理ソフトウェアは、アップグレード・キットを応用する必要があり、アップグレード・キット・アーカイブ・ファイルがローカル・マシンでアクセスできない場合には、自動的にアップグレード・キット・アーカイブ・ファイルをダウンロードすることができる。

#### アップグレードの開始

ソフトウェア・コンポーネントおよびアップグレード・キットを定義した後、アップグレードを行う者例えばデータベース管理者は、ソフトウェア・コンポーネントのアップグレードを開始できる。アップグレードを行う者例えばデータベース管理者は、まずソフトウェア・コンポーネントの要求されるバージョンを設定しなければならない。要求されるバージョンの設定には2つの方法がある。最初の方法は、コンポーネント画面を通じて行う。アップグレードを行う者例えばデータベース管理者はコンポーネント画面を使って、ソフトウェア・コンポーネントの要求されるバージョンを手操作で更新する。アップグレードを行う者例えばデータベース管理者は、画面を使って、1つまたはそれ以上のソフトウェア・コンポーネントの要求される最小または最大バージョンを修正する。

#### 【0045】

その代わりに、アップグレードを行う者例えばデータベース管理者は、アップグレード・キット画面を利用できる。アップグレードを行う者例えばデータベース管理者は、アップグレード・キット画面を使って、ソフトウェア・コンポーネントの要求されるバージョンを更新する。アップグレードを行う者例えばデータベース管理者は、アップグレード・キットを選択し、要求されるバージョンを設定するための機能を開始する。例えば、Siebel Remote の場合、アップグレードを行う者は、“Set Required Versions (要求されるバージョンを設定)” ボタンを押す。ボタンを押すと、アップグレードが影響する全てのソフトウェア・コンポーネントが探され、その最大バージョンが設定される。最大バージョンは、アップグレード・キットがインストールするバージョンに設定される。

#### 【0046】

アップグレードを行う者例えばデータベース管理者は、コンポーネント画面または同様の画面を使って、全てのユーザーに新しいソフトウェア・コンポーネン



ト情報を「リリース」する。アップグレードを行う者例えばデータベース管理者は、ソフトウェア・コンポーネントを選択して、“Release（リリース）” ボタンを押す。ボタンを押すと、選択されたソフトウェア・コンポーネントおよび関連ソフトウェア・コンポーネントに関する全ての情報が読み取られ、情報が確認され、情報が「コンパイル済み」のLONGカラムに書き込まれ、情報を全てのデータベースに複製するためにトランザクションがログされる。

#### アップグレード・キットの検出および応用

本発明のアップグレード・プログラムは、各コンポーネントの現在インストールされているバージョンを要求されるバージョンと比較することにより、自動的にアップグレードを検出する。クライアントは、スタートアップ時にソフトウェア・コンポーネントの「コンパイル済み」のLONGカラムから要求されるバージョンを読み取る。サーバーも、スタートアップ時にソフトウェア・コンポーネントの「コンパイル済み」のLONGカラムから要求されるバージョンを読み取る。

#### 【0047】

移動ユーザーおよび地域データベースの場合、アップグレードを行う者例えばデータベース管理者は、更新済みの要求されるバージョンを全てのリモート・データベースに分配する。ドッキング・クライアントおよび複製エージェントは、各同期化セッション前に、要求されるスキーマ・バージョンについてドッキング・サーバーに問い合わせる。同期化セッション中ソフトウェア・コンポーネントのコンパイル済みカラムに変化が生じた場合、ドッキング・クライアントおよび複製エージェントは、同期化セッション後に要求されるバージョンを読み取る。

#### 【0048】

バージョンのうちあるものが一致しない場合、アップグレードが要求され、プログラムは、1 つまたはそれ以上のアップグレード・キットを探して、下記の2段階でアップグレード・キットを応用する：

ローカル・マシンにアップグレード・キット・アーカイブ・ファイルをダウンロードする；

アップグレード・キットをローカル・マシンに応用するためにアップグレード・ウィザードを呼び出す。

**【0049】**

移動および地域データベース・インストレーションの場合、ドッキング・クライアントは、サーバー一般にはデータベース・サーバーからローカル・インボックス・ディレクトリにアップグレード・キット・アーカイブ・ファイルをダウンロードする。接続されるユーザーの場合には、クライアントは、ファイル・システムのアップグレード・キット・アーカイブ・ファイルをローカル・ディスクのテンポラリ・ディレクトリにダウンロードする。

**【0050】**

アップグレード・キットがローカル・マシンでアクセス可能になった後、アップグレード・ウィザードはアップグレードに応用される。アップグレード・ウィザードは、アップグレード・キット・アーカイブ・ファイルからファイルを抽出し、アップグレード・キットの各動作を順次実行する。エラーが生じた場合には、アップグレード・ウィザードはアップグレード・キット動作をロールバックして、エラーをユーザーに表示する。プログラムは、次回に、アップグレードを検出して、再びアップグレードを呼び出す。

**【0051】**

全てのアップグレード・キット動作を順調に完了した後、アップグレード・ウィザードはローカル・マシンからアップグレード・キット・ファイルを削除し、アップグレードを呼び出したプログラムをリスタートする。

**統合**

アップグレード・インストール・システムは、リモート・データベースを初期化して維持するためにパッケージ例えばSiebel Upgradeを使用する。アップグレード・ウィザードは、新しいリモート・データベースを初期化して、データベース再抽出を行い、データベース・スキーマ・アップグレードを処理する。

**【0052】**

リモート・データベースの抽出ーリモート・データベースを抽出する際、Database Extractはアウトボックス・ディレクトリにアップグレード・キット・アーカイブ・ファイルを作成する。このデータベース抽出アーカイブ・ファイルは、アップグレード・ウィザード・ドライバ・ファイル、およびローカル・データベ

ースを初期化するための下記の一つのアイテムから成る：

データベースの作成： データベース・テンプレート・ファイルをコピーする  
(移動ユーザーのため)

スナップショット・データ： 移動ユーザーのために可視データを含むデータベース・スナップショット・ファイルをインポートする。

#### 【0053】

Sql ファイル： ユーザーを作成しデータベースのデータを初期化する1組のSQL ファイルを実行する。

データベースの初期化

クライアント、複製エージェント、° キング・クライアントおよびスタンドアローンDbinit(データベース初期化) プログラムは、ローカル・データベースを初期化するためにDbinitを呼び出す。データベース初期化(Dbinit)は、要求される全てのバージョンが確実にローカル・マシンに存在し、移動クライアントのためにローカル・データベースを作成するようにする。

#### 【0054】

Dbinitは以下のステップを実施する：

サーバーへの接続： Dbinitは、ローカル移動クライアントの名前、移動クライアントのパスワードおよびリモート・サーバーのアドレスを入力するようユーザーに指示する(移動ユーザーは、アップグレードを行う者例えばMIS スタッフ、ベンダまたはデータベース管理者からこの情報を入手しなければならない)。データベース初期化は、リモート・サーバーに接続し、移動クライアントの名前、ユーザーおよびパスワードの妥当性を検査する。

#### 【0055】

バージョン・チェックの実施およびアップグレード・キットのダウンロード： Dbinitは、次に、必要とされるソフトウェア・コンポーネントが全てローカル・マシンにあることを確かめるためにバージョン・チェックを行う。Dbinitは、まず要求されるバージョン(すなわちコンパイルされた情報)をドッキング・サーバーからダウンロードする。Dbinitは、要求されるバージョンをローカル・マシンのバージョンと比較する。1つまたはそれ以上のバージョンが一致

しない場合、Dbinitは、全ての（一部ではなく全て？）アップグレード・キットに関する情報をダウンロードする。Dbinitは、次に、応用されるアップグレード・キットを探し、このアップグレード・キットをローカル・インボックス・ディレクトリにダウンロードする。

#### 【0056】

抽出スナップショット・ファイルのダウンロード： Dbinitはデータベース抽出スナップショット・ファイルをローカル・インボックス・ディレクトリにダウンロードする。

ドライバ・ファイルの作成： Dbinitは、アップグレード・キットをローカル・マシンに応用し、データベース抽出スナップショット・ファイルを使ってローカル・データベースを作成するために、アップグレード・ウィザード・ドライバ・ファイルを作成する。

#### 【0057】

アップグレード・ウィザードの実行： Dbinitは、アップグレード・キットを応用しローカル・データベースを初期化するために、アップグレード・ウィザードを呼び出す。

初期化が成功したら、アップグレード・ウィザードは、次に呼び出しプログラムをリスタートする。

#### クライアントからの自動初期化

移動ユーザーがローカル・データベースに対照してクライアントを実行するとき、クライアントは、ローカル・データベース・ファイルが存在するか否かをチェックする。ローカル・データベースが存在しない場合、クライアントは、ローカル・データベースを初期化したいかどうかを入力するようユーザーに指示する。初期化したい場合には、クライアントは、Dbinitを呼び出して、ローカル・データベースを作成し、全てのソフトウェア・コンポーネントが最新のものであるようにする。

#### バージョン・チェック

各同期化セッションの始めに、ドッキング・クライアントは、データベース抽出またはデータベース・スキーマ・アップグレードが未決であるか否かをチェッ

クする。これは、ドッキング・クライアントは、サーバー・データベースにトランザクションをアップロードできるかあるいはトランザクションを廃棄しなければならないかを知らなければならないため、ドッキング・セッションの始めにスキーマ・バージョン・チェックを行わなければならないためである。ドッキング・クライアントは、また、ローカル・データベースを再初期化するためにdbinitを実行しなければならないか否かも知らなければならない。

#### 【0058】

ドッキング・クライアントがドッキング・サーバーに接続した後、ドッキング・クライアントはドッキング・サーバーにサーバー・データベース・スキーマ・バージョンについて尋ねる。可能性のある結果は次の3つである：

大幅なスキーマ変更： ローカル・データベースとサーバー・データベースの主要なおよび重要でないバージョンが一致しない場合、データベース再初期化が「要求される」。主要なまたは重要でないスキーマ・バージョンの変更は、サーバー・データベースが大幅なスキーマ変更を受けたことを暗示する。ドッキング・クライアントは、全ての未決トランザクションを廃棄する。ドッキング・クライアントは、ローカル・データベースを初期化してソフトウェア・コンポーネント（もしあれば）をアップグレードするために、移動ユーザーに知らせて、アップグレード・ウィザードを呼び出す。データベース抽出が利用不能である場合、ドッキング・クライアントはエラー・メッセージを表示して、ドッキング・クライアントを再抽出するためにアップグレードを行う者例えばデータベース管理者に連絡するよう、移動ユーザーに求める。

#### 【0059】

未決dbinit： 大幅なスキーマ変更がなく、アップグレードを行う者例えばデータベース管理者がドッキング・クライアントを再抽出した場合、ドッキング・クライアントは、ローカル・データベースを再初期化しなければならない。ドッキング・クライアントは、全ての未決トランザクションをセーブして、dbinit.dxと呼ばれるインボックス・ディレクトリのファイルにアップロードする。ドッキング・クライアントは、新しいローカル・データベースを作成し新しいソフトウェア・コンポーネントをアップグレードするために、アップグレード・ウィザ

ードを呼び出す。ローカル・データベースの初期化後、アップグレード・ウィザードは、dbinit.dx にセーブされた未決トランザクションをローカル・データベースに応用するためにデータ・マージ・ユーティリティ (dmutl.exe) を呼び出す。Dmutl.exe は、トランザクションをローカル・データベースに応用し、新しいトランザクションをログして、移動ユーザーが全てのトランザクションを再入力したかのようにサーバー・データベースにアップロードする。

#### 【0060】

微細なスキーマ変更： 大幅なスキーマ変更および未決dbinitがない場合、ドッキング・クライアントは、ドッキング・クライアントの初期化を進める。最終的に、ドッキング・クライアントは、スキーマ・ソフトウェア・コンポーネントの要求されるバージョンをアップグレードするためにトランザクションを応用する。この更新をきっかけとして、ドッキング・クライアントはバージョン・チェックを行い、最終的にアップグレード・ウィザードはローカル・データベースをアップグレードするためにアップグレード・キットを応用する。

#### 地域データベースの初期化

地域データベースにおいてアップグレードを行う者例えばデータベース管理者はスタンドアローン・データベース初期化プログラムを実行して、地域データベースを初期化する。初期化プログラムは、リモート・サーバーに接続して、アウトボックス・ディレクトリからデータベース抽出アーカイブ・ファイルをダウンロードする。次に、プログラムは、地域データベースを初期化するためにアップグレード・ウィザードを呼び出す。

#### 地域データベースの置き換え

HQサーバーにおいてアップグレードを行う者例えばデータベース管理者が既存の地域データベースのためにデータベース抽出を実行する場合、地域データベースにおいてアップグレードを行う者例えばデータベース管理者は、地域データベースを再初期化する必要がある。各同期化セッションの始めに、複製エージェントは、地域データベースについてデータ抽出アーカイブ・ファイルが未決か否かをチェックする。未決の場合には、複製エージェントは、アップグレードを行う者例えばデータベース管理者に知らせて、シャットダウンする。アップグレード

を行う者例えばデータベース管理者は、地域データベースを初期化するために手操作でスタンドアローン・データベース初期化プログラムを呼び出さなければならない。

#### 【0061】

スキーマ・アップグレード・キットは、以下のアイテムのうち1 つまたはそれ以上により構成される：

スキーマ変更を加えるためのDDL ファイル

シード・データをインストールするためのDAT ファイル

レポジトリをインストールするためのRepimexpファイル

特殊なアップグレード・ステップの実行、データ変換などのためのSQL スクリプト

S \_\_APP \_\_VER テーブルのバージョン番号を更新するためのSQL スクリプト

ソフトウェア・コンポーネントを更新するためのファイル： これには、ソフトウェア・コンポーネント情報の追加、削除および更新が含まれる。最低、ファイルはコンポーネントの要求されるバージョンを更新しなければならない（例えば、新しいスキーマに対してTools の新しいバージョンを使用しなければならない）。

#### 【0062】

アップグレード・キットをインストールするためのファイル。例えば、Tools、Serverなどの新しいバージョンをインストールするためのアップグレード・キット。

カスタマ・サイトでのスキーマのインストール

アップグレードを行う者例えばデータベース管理者は、CDまたはその他の媒体を受け取り、アップグレード・キットをテスト/ 構成データベースに応用する。アップグレード・キットを応用する前に、アップグレードを行う者例えばデータベース管理者は、まず全てのプログラムを、接続されるユーザーもサーバー・インスタンスもシャットダウンする。次に、アップグレード・キットをテスト・データベースに応用するためにアップグレード・ウィザードを実行する。

#### 【0063】

アップグレード・キットは、以下のステップを行う：

スキーマ変更を加える

シード・データをインポートする

新しいレポジトリをインポートするためにrepimexp.exeを実行する

特殊なアップグレード・ステップの実行、データ変換などのためにSQL スクリプトを実行する

バージョン番号を更新する

ソフトウェア・コンポーネント・テーブルを更新するためにファイルを実行する

定義されたアップグレード・キットをインストールするためにファイルを実行する

アップデート後のデータベースのアクセス

この時点で、テスト/ 構成データベースは新しいスキーマを持ち、ソフトウェア・コンポーネントの要求されるバージョンは、新しいバージョンに設定される。サーバー・データベースに接続するプログラムは、要求される新しいバージョンになるために1 つまたはそれ以上のアップグレード・キットを応用する必要がある。例えば、アップグレードを行う者例えばデータベース管理者がサーバー・データベースにログインする場合、ログインに使用されるソフトウェアはバージョンの不一致を検出し、アップグレードしようとするかアップグレードを呼び出さなければならない。アップグレードを行う者例えばデータベース管理者は、新しいTools バージョンにアップグレードするために次の2 つの選択肢を持つ：アップグレード・キットが利用可能であれば、アップグレード・キットをダウンロードして、これをローカル・マシンに応用する。

#### 【0064】

CD-ROMから新しいバージョンをインストールする。その後、サーバー・データベースと対照させてバージョンを使用する。ローカル・マシンにソフトウェアをインストールするのが初めての場合、アップグレードを行う者例えばデータベース管理者はこの選択肢を選ばなければならない。

アップグレードを行う者例えばデータベース管理者がプログラムにアップグレ



ードを行わせたい場合、各種のソフトウェア・コンポーネントをアップグレードするためにアップグレード・キットを記録することができる。そうでなければ、CD-ROMからインストールすることによって各マシンを手操作でアップグレードすることができる。

#### カスタマイズおよびテスト

アップグレードを行う者例えばデータベース管理者は、下記の3つの方法でローカル・インストレーションをカスタマイズすることができる：

新しい拡張テーブル、拡張カラムまたはインデクスなどデータベース拡張  
新しいCDF ファイル、CFG ファイルまたはリポート・ファイルなど構成変更  
Microsoft Word、Adobe Acrobat などの新しいバージョンなど第三者ソフトウェア変更

カスタマは、この種のカスタマイズを作成し分配するための充分に定義されたプロシージャに従う。このカスタマイズは、全て大幅なスキーマ変更を生じないので、データベースの再抽出を必要としない。

#### 【0065】

ステップ1： サーバー・データベースに構成変更を加える。アップグレードを行う者例えばデータベース管理者は、そのサイトのためにエンドユーザー・インストレーションをカスタマイズし、カスタマのレポジトリを新しいレポジトリとマージするためにプログラムを使用する。

バージョンまたはスキーマ・アップグレード中構成変更を行うために、新しいCDF ファイルを生成する必要がある。これは、CFG ファイルをカスタマイズするためのエディタおよびレポートをカスタマイズするためのMicrosoft AccessなどのDBMSプログラムを使って行われる。Microsoft Help Editor などを使ってHelp ファイルをカスタマイズすることができる。

#### 【0066】

アップグレード・キットの中のプログラムは、バージョンを比較して、次のスタートアップ時にアップグレードを応用する必要性を自動的に検出する。本発明の方法、装置、システムおよびプログラム製品は、接続されるユーザーをアップグレードする。同様に、サーバーは、アプリケーション・サーバーをアップグレ

ードし、また存在する場合には、ドッキング・クライアントはローカル・データベースをアップグレードする。複製エージェントは地域データベースをアップグレードする。

【0067】

本発明の方法、装置、システムおよびプログラムは、必要とされるアップグレード・キットをダウンロードし、アップグレードをローカル・マシンに応用する。データベース・アップグレードが主要なまたは重要でないスキーマ・バージョンを変更した場合、本発明のアップグレード方法、システム、装置およびプログラム製品は、自動的に、移動または地域データベースを再初期化する。

【0068】

本発明の別の態様は、ソフトウェア・コンポーネントの要求されるバージョンの自動設定である。その代わりに、アップグレードを行う者例えばデータベース管理者は、要求されるバージョンを更新して、1つまたはそれ以上のソフトウェア・コンポーネントのアップグレードを開始することができる。ネットワーク環境において、アップグレードを行う者例えばデータベース管理者は、要求されるバージョンのリストに変更を加える際非常に慎重に行わなければならない。なぜなら、このファイルの更新により、あらゆるロケーションのプログラムがアップグレードを呼び出す可能性があるからである。

【0069】

ほとんどの場合、アプリケーション・サーバーは自己アップグレードを行う。すなわち、アップグレードを行う者例えばデータベース管理者は、アップグレードを検出し応用するために介入する必要がある。しかし、例外がある。アップグレード・キットがユーザーの介入を要求する場合、アップグレード・ソフトウェアは、アップグレードを行う者例えばデータベース管理者が入力を行うまで待たなければならない。

【0070】

同様に、例えば地域データベースについてデータベース抽出が未決の場合、サーバーは地域データベースへの全てのアクセスをシャットダウンし、トランザクション・マージャが移動クライアントから全ての未決. dx ファイルを順調にマー

ジしたことを確認し、地域データベースを再初期化するためにデータベース初期化プロセスを開始しなければならない。

#### 【0071】

本発明のソフトウェア・アップグレード・プログラムは、全てのマシンに正確なソフトウェア・コンポーネントのバージョンを維持するために必要に応じて、アップグレードを自動的に検出し、応用する。ユーザーは、アップグレードを手操作で探して、応用する必要はない。特にLAN またはWAN に接続されるユーザーは、アップグレードに苦痛を感じない。クライアントは、必要に応じて、自動的にアップグレードを検出して、応用する。アップグレードが検出されると、クライアントは、アップグレードするソフトウェア・コンポーネントをユーザーに通知し、ユーザーにアップグレードを勧め、アップグレードをスタートするようユーザーを促す。ローカル・ディスクにアップグレード・キットがまだ存在しない場合には、システムは適当なアップグレード・キットをローカル・ディスクに自動的にダウンロードする。

#### 【0072】

アップグレード・キットがローカル・マシンで利用可能になった後、クライアントは、アップグレードを応用するためにアップグレード・ウィザードを呼び出す。アップグレード・ウィザードはアップグレード・キット・アイテムを表示して、アップグレードをスタートするようユーザーを促す。アップグレード・ウィザードは、アップグレード・キット・アイテムの処理をスタートし、定期的に進行についてユーザーに知らせる。アップグレード・ウィザードが完了したら、アップグレード・ウィザードはオリジナルのプログラムをリスタートする。

#### 【0073】

移動ユーザーも、アップグレードに苦痛を感じない。移動ユーザーは、明示的にデータベース初期化を呼び出す必要がない。ローカル・データベースが利用可能でない場合にローカル・データベースに対してクライアントをスタートしようとする、自動的にデータベース初期化を呼び出して、ローカル・データベースが作成される。Dbinitも、ローカル・マシンの全てのソフトウェア・コンポーネントを最新のものにする。

**【0074】**

移動ユーザーは、アップグレード・キット画面にナビゲートし、次の同期化セッションでアップグレード・キット・アーカイブ・ファイルをダウンロードするよう要求することによって、アップグレード・キットを必要とするより以前にこれをダウンロードすることができる。

本発明のアップグレード・ソフトウェアは、各同期化セッションの始めにバージョン・チェックを行う。アップグレードが検出されると、クライアントは、アップグレードするソフトウェア・コンポーネントをユーザーに通知し、ユーザーにアップグレードを勧め、アップグレードをスタートするようユーザーを促し、ローカル・ディスクにアップグレード・キットがまだ存在しない場合には、適当なアップグレード・キットをローカル・ディスクに自動的にダウンロードする。

**【0075】**

アップグレード・キットがローカル・マシンで利用可能になった後、クライアントは、アップグレードを応用するためにアップグレード・ウィザードを呼び出す。アップグレード・ウィザードはアップグレード・キット・アイテムを表示して、アップグレードをスタートするようユーザーを促す。アップグレード・ウィザードは、アップグレード・キット・アイテムの処理をスタートし、定期的に行進についてユーザーに知らせる。アップグレード・ウィザードが完了したら、アップグレード・ウィザードはオリジナルのプログラムをリスタートする。

**【0076】**

再初期化が必要な場合、ドッキング・クライアントは、ローカル・データベースを再初期化するためにDbinitを呼び出す。

本発明の1つの実施例においては、コンポーネント画面は、アップグレードを行う者例えばデータベース管理者に、ソフトウェア・コンポーネントを定義させ、ソフトウェア・コンポーネントについて要求されるバージョンを指定させる。この画面は、また、アップグレードを行う者例えばデータベース管理者に、ソフトウェア・コンポーネント間の依存性を指定させる。例えば、クライアント・ソフトウェア・コンポーネントは、クライアントがスタートアップする前にスキーマ・コンポーネントがすでにインストールされていることを要求する。

**【0077】**

コンポーネント画面は次の2つのアプレットを持つ： ソフトウェア・コンポーネントの名前および要求されるバージョンを定義するソフトウェア・コンポーネント・アプレット、および現ソフトウェア・コンポーネントを使用するために必要とされる他のソフトウェア・コンポーネントのリストを定義する必要とされるソフトウェア・コンポーネント・アプレット。後者は、ソフトウェア・コンポーネント・アプレットの子である。

**【0078】**

ソフトウェア・コンポーネント・アプレットは、アップグレードを行う者例えばデータベース管理者に、ソフトウェア・コンポーネントの属性を入力させる。このアプレットは、以下のフィールドのうち1つまたはそれ以上を持つことができる：

名前(必須)： ソフトウェア・コンポーネントの名前。

**【0079】**

コメント(任意)： ソフトウェア・コンポーネントの説明。

最小バージョン(任意)： このソフトウェア・コンポーネントを使用するためにマシンにインストールされなければならない最小バージョン。NULLの場合、ソフトウェア・コンポーネントは最小バージョンを持たない。

最大バージョン(任意)： ソフトウェア・コンポーネントを使用するためにマシンにインストールされなければならない最大バージョン。NULLの場合、ソフトウェア・コンポーネントは最大バージョンを持たない。アプレットは、最小バージョンが最大バージョンより小さいかこれに等しいことを確認する。

**【0080】**

ロケート方法(任意)： ソフトウェア・コンポーネントのインストール・ロケーションを得るために使用する方法。これはピックリストである。

ロケート情報(任意)： ロケート方法の追加情報。

バージョン方法(任意)： ソフトウェア・コンポーネントの現在インストールされているバージョンを得るための方法。これはピックリストである。

**【0081】**

バージョン情報(任意) : バージョン方法の追加情報。

ソフトウェア・コンポーネント・アプレットは、アップグレードを行う者例えばデータベース管理者に、ソフトウェア・コンポーネントのために必要とされるソフトウェア・コンポーネントを定義させる。このアプレットは、以下のフィールドを持つ：

必要とされるソフトウェア・コンポーネント(必須) : 必要とされるソフトウェア・コンポーネントの名前。

#### 【0082】

コメント(任意) :

スタートアップ・フラグ(必須) : 必要とされるソフトウェア・コンポーネントがソフトウェア・コンポーネントをスタートアップするために必要かどうかを指定する。必要とされるソフトウェア・コンポーネントの中には、ソフトウェア・コンポーネントによって使用されるがスタートアップのためには必要とされないものがある。例えば、クライアントは、ユーザーがCorrespondenceを使用する場合にはMicrosoft Wordを必要とするが、クライアントをスタートアップするときMicrosoft Wordは必要とされない。

#### 【0083】

アップグレード・キット・アイテム引数画面。各アップグレード・キット・アイテム・タイプの有効引数のリストを定義するためにこの画面を使用することができる。この画面はアプレットを1つ持つ：

アップグレード・キット・アイテム・タイプ引数アプレット : このアプレットは、アップグレードを行う者例えばデータベース管理者にアップグレード・キット・アイテム・タイプの有効引数を入力させる。このアプレットは以下のフィールドを持つ：

アイテム・タイプ(必須) : 応用されるアップグレード・キット・アイテムのタイプ。Error! Reference source not found は、アップグレード・キット・アイテムの有効タイプおよびそれぞれの引数を示す。これはピックリストである。

#### 【0084】

順序（必須）： アップグレード・ウィザードがアップグレード・キット・アイテムを応用するために引数をパスする順序を示す整数。

必要とされるフラグ（必須）： 引数が要求されるか否かを指定する。

コメント（任意）： アップグレード・キット・アイテム・タイプ引数の説明。

#### 【0085】

デフォルト値（任意）： 引数のデフォルト値。

アップグレード・キット・アプレット。このアプレットは、アップグレードを行う者例えばデータベース管理者に、アップグレード・キットの属性を入力させる。このアプレットは以下のフィールドを持つ：

名前（必須）： アップグレード・キットの名前。

#### 【0086】

タイトル（必須）： アップグレード・キットのユーザーフレンドリーな名前。アップグレード・ウィザードは、アップグレード・ウィザード対話ボックスのタイトル・バーにこのストリングを表示する。この値はアップグレード・キットの名前をデフォルトとする。

コメント（任意）： アップグレード・キットの説明。アップグレード・ウィザードは、アップグレード・ウィザードのスタートアップ時にこの説明を表示する。

#### 【0087】

状態（必須）： アップグレード・キットの現状態。アップグレード・キットの定義の際状態は「ペンディング（未決）」に設定される。アップグレードを行う者例えばデータベース管理者が「リリース」ボタンを押すと、状態は「アクティブ（活動中）」に変化する。アップグレードを行う者例えばデータベース管理者が状態を変更し、アップグレード・キットで記録削除動作を選択すると、状態は「イナクティブ（不活動）」に変化する。

#### 【0088】

アップグレード・キット・アーカイブ・ファイル（任意）： 付属のアップグレード・キット・アーカイブ・ファイルの属性。これは、ファイル添付ビジネ

ス・コンポーネントおよびフレームが使用する属性である。移動ユーザーはこのアプレットにナビゲートして、次の同期化セッションにリモートがアップグレード・キット・アーカイブをダウンロードするよう要求することができる。

#### 【0089】

アップグレードを行う者例えばデータベース管理者がアップグレード・キットを「リリース」した後、移動ユーザーまたは接続されるユーザーは、アップグレード・キットをローカル・マシンにダウンロードしてこれを応用することができる。ユーザーは、このアプレットにナビゲートして、アップグレード・キット上でダブルクリックする。ローカル・マシンで（すなわち移動ユーザーが）アップグレード・キット・アーカイブ・ファイルにアクセスできない場合、アプレットは、次の同期化セッションでアップグレード・キット・アーカイブ・ファイルをダウンロードするよう要求したいか否かユーザーに尋ねる。アップグレード・キットがローカル・マシンですでにアクセス可能な場合、アプレットは、ローカル・マシンにアップグレード・キットを応用するためにアップグレード・ウィザードを呼び出す。アップグレード・ウィザードは、アップグレードをスタートする前にクライアントが終了（エグジット）するのを待つ。

#### 【0090】

アップグレード・キット・アイテム・アプレット。このアプレットは、アップグレードを行う者例えばデータベース管理者にアップグレード・キットのアイテムを入力させる。このアプレットは以下のフィールドを持つ：

順序（必須）： アップグレード・キット・アイテムの実行順序を示す整数。

タイトル（必須）： アップグレード・キット・アイテムのユーザー・フレンドリーなタイトル。アップグレード・ウィザードは、アップグレードにおいて行われるアイテムのリストにこのストリングを表示する。

#### 【0091】

コメント（任意）： アップグレード・キット・アイテムの説明。アップグレード・ウィザードは、ユーザーがアップグレード・キット・アイテムをハイライトするときこの説明を表示する。

ディスク・スペース推定（任意）： アップグレード・キット・アイテムをあ



るマシンに应用するために必要なディスク・スペース量。アップグレードを行う者例えばデータベース管理者は、この推定量を示す。推定がNULLの場合、アップグレード・ウィザードは、予め定められたいくつかのルールに基づいてディスク・スペース推定を行う。

#### 【0092】

アイテム・タイプ( 必須) : 応用されるアップグレード・キット・アイテムのタイプ。Error ! Reference source not foundは、アップグレード・キット・アイテムの有効タイプおよびそのそれぞれの引数を示す。

アップグレード・キット・アイテム引数アプレットは、アップグレードを行う者例えばデータベース管理者にアップグレード・キット・アイテムの引数を入力させる。各アップグレード・キット・アイテム・タイプについての有効引数のリストは、S \_\_UPG \_\_ITARG テーブルに記憶される。このアプレットは、アップグレード・キット・アイテムとアップグレード・キット・タイプ引数の間の交差記録を作成する。このアプレットは以下のフィールドを持つ：

名前( 必須) : 引数の名前。

#### 【0093】

コメント( 任意) : アップグレード・キット・アイテムの説明。アップグレード・ウィザードは、ユーザーがアップグレード・キット・アイテムをハイライトするときこの説明を表示する。

引数値( 必須)

ファイル属性( 任意) : アップグレード・キット引数に付属するファイルの属性。アップグレードを行う者例えばデータベース管理者は、アップグレード・キット・アイテムにファイルを添付する。

#### 【0094】

アップグレード・キット・コンポーネント・アプレットは、アップグレードを行う者例えばデータベース管理者に、どのソフトウェア・コンポーネントがこのアップグレードのために必要とされるかを定義させる。例えば、特定のローカル・マシンにバージョン3.1.0 以上をすでにインストールしていなければ、バージョン3.1.2 をインストールすることはできない。このアプレットは、アップグレ

ードを行う者例えばデータベース管理者に、アップグレード・キットが影響を及ぼすソフトウェア・コンポーネントも指定させる。

【0095】

このアプレットは、以下のフィールドを持つ：

ソフトウェア・コンポーネント（必須）： このアップグレード・キットの影響を受けるソフトウェア・コンポーネント。

最小インストール・バージョン（任意）： このフィールドは、このアップグレード・キットをインストールするために要求されるソフトウェア・コンポーネントの最小バージョンを記憶する。アップグレード・キットがこのソフトウェア・コンポーネントを必要としない場合には、このフィールドは、NULLにしておくことができる。

【0096】

最大インストール・バージョン（任意）： このフィールドは、このアップグレード・キットをインストールするために要求されるソフトウェア・コンポーネントの最大バージョンを記憶する。アップグレード・キットがこのソフトウェア・コンポーネントを必要としない場合には、このフィールドはNULLにしておくことができる。

【0097】

インストールされる新しいバージョン（任意）： このフィールドは、このアップグレード・キットがあるマシンに応用される場合ソフトウェア・コンポーネントの新しいバージョンを記憶する。アップグレード・キットがソフトウェア・コンポーネントの新しいバージョンをインストールしない場合には、このフィールドはNULLにしておくことができる。

【0098】

このアプレットの対話は、最小バージョンが最大バージョンより小さいかこれに等しいことを確認する。また、新しいバージョンが最大バージョンより大きいかにこれに等しいことも確認する。

アップグレード・キット・ウィザードは、カスタマがもっとも頻繁に必要とするアップグレード・キットをアップグレードを行う者例えばデータベース管理者

が作成するのを助けるために1組の定義済み対話を含んでいる。これは、以下のエレメントを1つまたはそれ以上含むことができる：

画面1： アップグレード・キット・タイプの選択

データベース拡張： データベース拡張を応用するためにアップグレード・キットを作成する。アップグレード・キットは、2つのアップグレード・キット・アイテム、すなわちデータベース・スキーマ変更を応用するためのDDL ファイル、およびレポジトリ・データ・ファイルをリモート・データベースに分配するためのDAT ファイル、によって構成される。レポジトリ・データ・ファイルは、辞書共通API を初期化するために必要なレポジトリ・テーブルのサブセットを含んでいる。

【0099】

CDF ファイル： 新しいCDF ファイルを全てのユーザーに分配するためにアップグレード・キットを作成する。

リポート・ファイル： 新しいリポート・ファイルを全てのユーザーに分配するためにアップグレード・キットを作成する。

ヘルプ・ファイル： 新しいヘルプ・ファイルを全てのユーザーに分配するためにアップグレード・キットを作成する。

【0100】

サンプル・データベース： 新しいサンプル・データベースを全てのユーザーに分配するためにアップグレード・キットを作成する。

第三者ソフトウェア： 第三者ソフトウェアを全てのユーザーに分配するためにアップグレード・キットを作成する。

画面2： フルまたはパッチ・インストール

フル・インストール

パッチ・インストール

画面3： バージョン情報

要求される最小バージョンを入力： フル・インストールの場合NULLをデフォルトとする。その他の場合？をデフォルトとする。

【0101】

要求される最大バージョンを入力： フル・インストールの場合NULLをデフォルトとする。その他の場合\_をデフォルトとする。

新しいバージョンを入力： ソフトウェア・コンポーネントの現最大バージョン+1をデフォルトとする。

対話は最小バージョンが最大バージョンより小さいかこれに等しいことを確認する。また、新しいバージョンが最大バージョンより大きいかこれに等しいことも確認する。

#### 【0102】

画面4： ファイル

ソース・ファイルを入力： フル・インストールの場合、アップグレード・キットに関するファイルを含むディレクトリを選択する。アップグレード・キット・ウィザードはディレクトリおよびサブディレクトリの中の全てのファイルをアップグレード・キットに追加する。

#### 【0103】

パッチ・インストールの場合、アップグレード・キットに追加する1 つまたはそれ以上のファイルを選択する。

ファイルの宛て先ディレクトリを入力。

画面5： 確認

新しいアップグレード・キットに関する情報を検討し、新しいアップグレード・キットを作成するためにはOKを押す。そうでなければ、Cancelを押して打ち切る。

#### 【0104】

本発明のシステム、方法、装置およびプログラム製品の主要な構成要素はアップグレード・ウィザードである。アップグレード・ウィザードの主要な機能は、アップグレード・キットを応用することである。本発明の望ましい実施例において、アップグレード・ウィザードは非常に単純なユーザー・インターフェイスを持っている。アップグレード・ウィザードが最初にスタートアップするとき、これはアップグレード・キットのタイトルおよび説明をユーザーに表示する。ユーザーがOKまたはその他の同様のボタンを押すと、アップグレード・ウィザードは

アップグレード・キット・アイテムのリストをユーザーに表示する。アップグレード・ウィザードは、アップグレード・キット・アイテムのタイトルを表示する。ユーザーが特定のアップグレード・キット・アイテムをハイライトすると、アップグレード・ウィザードは、ボトム・フレームに選択されたアイテムの説明を表示する。

#### 【0105】

ユーザーがアップグレード・キット・アイテムを検討して、START またはその他の同様のボタンを押すと、アップグレード・ウィザードはアップグレード・キット・アイテムの各々を順番に実行する。バー・インディケータがアップグレードの推定進行を示すことができる。アップグレード・ウィザードは、また、アイテムが完了した後アップグレード・キット・アイテムをハイライトすることもできる。

#### アルゴリズムおよびプロシージャ

バージョン・チェック機能： ソフトウェア・コンポーネントのバージョンをチェックするために、アップグレード・ウィザードは、データベースから要求されるバージョンをまたソフトウェア・コンポーネント自身から実際のバージョンを入手する。どのコンポーネントがチェックを必要とするかを判定するために、アップグレード・ウィザードはベース・コンポーネントから始めて、必要とされるコンポーネント・ネットワークを通り抜けて、必要とされるコンポーネントのリストを得る。

#### 【0106】

アップグレード・ウィザード・プログラムは、バージョン・チェックを行うための機能を呼び出す。コール・ルーチンは、「コンパイル済み」のコンポーネント情報を送る。この機能は、古くなったソフトウェア・コンポーネントおよび古くなった各ソフトウェア・コンポーネントの現バージョンおよび要求されるバージョンのリストを返す。

#### 【0107】

この機能は、「コンパイル済み」のコンポーネント情報に記載される各ソフトウェア・コンポーネントのバージョンをチェックしなければならない。まず、こ

の機能は、コンポーネントをロケートしなければならない。この点に関して、Error! Reference source not found は、ソフトウェア・コンポーネントのロケートするための方法のリストを示す。ロケートし終わったら、この機能は、コンポーネントのバージョンをチェックし、コンポーネントをアップグレードできるかどうか判定できる。CDF ファイルは、バージョン番号を記憶する。アップグレード・キット・ウィザードは、CDF ファイルに値を書き込む。

#### 【0108】

CFG ファイル： 新しいバージョン属性を [] セクションに追加する。アップグレード・キット・ウィザードは、値をCFG ファイルに書き込む。

どのアップグレードを実行するか決定： アップグレード・キット・コンポーネント・テーブルはアップグレード・キットに関するコンポーネントおよびバージョン情報を記憶する。このテーブルは、アップグレード・キットがどのコンポーネントをインストールし、どのコンポーネントがアップグレードによって必要とされるかを示す。どのアップグレード・キットが応用されるかの検出は、自明ではない場合がある。

#### 【0109】

本発明の方法、装置、システムおよびプログラム製品は、必要なアップグレード・キットを探すために帰納的アルゴリズムを使用する。アルゴリズムは下記のとおりである：

If current\_\_versions=desired\_\_versions (現バージョン=所望のバージョンなら)

Call callback with current upgrade list (現アップグレード・リストと一緒にコールバックを呼び出し)

Return TRUE; (TRUEに復帰)

If upgrade list is full (アップグレード・リストがフルなら)

Return TRUE; (TRUEに復帰)

For each upgrade (各アップグレードごとに)

Do

If upgrade already in upgrade list(アップグレードがすでにアップグレ

ード・リストにある場合)

Continue; (継続)

If upgrade can't be run now(アップグレードをいま実行できない場合、すなわちupgrade \_\_required\_\_versions!=current \_\_versions)

Continue;(継続)

Add upgrade to the list (アップグレードをリストに追加)

Apply upgrade information to current\_\_versions (アップグレード情報を現バージョンに応用)

Recurse (再帰)

Remove upgrade from the list (アップグレードをリストから除去)

Remove upgrade information from current \_\_versions (アップグレード除法を現バージョンから除去)

Done (終了)

Return TRUE; (TRUEに復帰)

このアルゴリズムは、現在インストールされているバージョンから要求されるバージョンにするためのアップグレード・キットのリストを作成する。パスが複数ある場合、この機能は最少アップグレード・キットのパスを選択する。

#### 【0110】

アップグレード・ウィザード： アップグレード・ウィザードは、単純なスタンドアローン・プログラムであり、その唯一の機能は、アップグレード・キットをローカル・マシンに応用することである。アップグレード・ウィザードは、データベースに接続せず、データベースを実行する必要がない。アップグレード・ウィザードは、全てのパラメータを読み取り、単純なファイルに状態情報を記憶する。アップグレード・ウィザードは、トランザクションをマージする能力を持つ必要はない。アップグレード・ウィザードは、Dbinitのインプリメンテーション部分（転送部分でなく）実行することができる。

#### 【0111】

アップグレード・ウィザードの実行： アップグレード・ウィザードは2 つの

タイプのファイル、すなわち状態ファイルおよびアップグレード・ドライバ・ファイルを読み取る。どちらのタイプのファイルも人間に読み取り可能なので、必要であればサポートがファイルを「フィックス」できる。

状態ファイル： 状態ファイルは、実行する必要があるアップグレードのリストおよびアップグレードの状態を含んでいる。状態リストは、未決または進行中のアップグレードがなければ空である。クライアントおよびサーバー・プログラムはスタートアップ中状態ファイルをチェックする。状態ファイルが空でなければ、クライアントおよびサーバー・プログラムは、アップグレードを再開するためにアップグレード・ウィザードを呼び出す。

#### 【0112】

ドライバ・ファイル： アップグレード・ドライバは、アップグレードのために実行する必要があるアップグレード・キット・アイテムのリストを含んでいる。Error! Reference source not found は、有効なアップグレード・キット・アイテムのリストを示す。

スタートアップ時に、アップグレード・ウィザードは、ドライバ・ファイルを読み取り、アップグレードを応用するのに十分なディスク・スペースがあることを確認する。アップグレード・ウィザードは、各アップグレード・キット・アイテムについて提供されるディスク・スペース推定量を使用する。アップグレードを行う者例えばデータベース管理者が推定値を示さない場合、アップグレード・ウィザードは、ある種のDun アルゴリズムを使ってディスク・スペースを推定する。アップグレード・ウィザードは、ローカル・マシンにアップグレード・キットを応用するために十分なディスク・スペースがない場合には、ユーザーに知らせる。

#### 【0113】

アップグレード・ウィザードは次に状態ファイルを読み取る。状態ファイルが空でない場合、アップグレード・ウィザードは、まず完了済みのアップグレード・キット・アイテムを逆順に実行することによって、全ての変更をロールバックする。

アップグレード・ウィザードは、ローカル・マシンにアップグレード・キット



を応用する時間を推定し、ユーザーがアップグレードをスタートするのを待つ。アップグレード・ウィザードは、動作を処理しながら状態ファイルを更新する。ウィザードは変更をディスクに書き出す（ffshおよび\_\_commitを使って）。アップグレード・ウィザードは、各動作を実行する前にバックアップ・ディレクトリにデータをセーブする。xxx は、各アップグレード・キット・アイテム・タイプごとにウィザードが行うバックアップ・ステップを示す。例えば、データベース関係の動作を実行する前に、アップグレード・ウィザードは、ローカル・データベースをバックアップする。実際にはこれによって回復可能性が向上するとは思われないが、害にはならない。アップグレードが順調に完了したら、アップグレード・ウィザードは全てのバックアップ・ファイルを削除する。また、アップグレード状態ファイルも削除する。

#### 【0114】

アップグレード・ウィザードのアップグレード： アップグレード・ウィザードは他のプログラムへの依存性は持たない。アップグレード・ウィザードはCFGファイルまたはDLLをロードしない。アップグレード・ウィザード自体のアップグレードをサポートするために、アップグレード・ウィザードを呼び出すプログラムは下記のステップに従う：

実行可能アップグレード・ウィザードの「レディ状態」バージョンが存在するか否かチェックする。

#### 【0115】

「レディ状態」バージョンが存在する場合、既存の実行可能アップグレード・ウィザードを削除して、「レディ状態」バージョンを現実行可能アップグレード・ウィザードに改名する。

実行可能アップグレード・ウィザードが存在しない場合、ソース・アップグレード・ウィザードを実行可能アップグレード・ウィザードに改名する。

#### 【0116】

アップグレード・ウィザードは、アップグレードを終了すると、ソース・アップグレード・ウィザードをレディ状態アップグレード・ウィザードに改名する。アップグレード状態ファイルが空で、ユーザーが状態ファイルのフォーマッ

トを変更できるときに、これが行われるのが望ましい。その代わりに、次のアップグレードが新しいアップグレード・ウィザードの機能および新しいアップグレード・ウィザード・ドライバ・ファイルのシンタックスを利用できるように、アップグレードされた実行可能アップグレード・ウィザードを生成したアップグレードの直後に、改名を行なうことができる。

#### 【0117】

このアルゴリズムにより、たとえアップグレード・ウィザードの新しいバージョンをインストールした後にリスタートされても、アップグレード全体を通じて同じアップグレード・ウィザードが使用される。

バージョン・チェック機能の呼び出し： 各プログラムは、予め決められた時点でバージョン・チェック機能呼び出す。

#### 【0118】

クライアント：

スタートアップ時に、クライアントに関してバージョン・チェックを行う。クライアントは、S \_\_UPG \_\_COMPテーブルの「コンパイル済み」LONGカラムから要求されるバージョンを読み取り、情報をバージョン・チェック機能に送る。

サーバー：

スタートアップ時に、サーバーは、バージョン・チェックを行うために新しいサーバー・コンポーネントサーバー・バージョン・チェッカーをスタートする。サーバー・バージョン・チェッカーは、S \_\_UPG \_\_COMPテーブルの「コンパイル済み」LONGカラムからサーバー・コンポーネントについて要求されるバージョンを読み取り、情報をバージョン・チェック機能に送る。

#### 【0119】

ドッキング・クライアントおよび複製エージェント：

各同期化セッション・スタート時に、ドッキング・クライアント（または複製エージェント）は、DRL メッセージを通じてサーバー・データベースのスキーマ・バージョンを得る。ドッキング・サーバーは、サーバー・データベースS \_\_APP \_\_VER テーブルを読み取り、サーバー・データベース・スキーマ・バージョンをドッキング・クライアントに送る。サーバー・データベースとローカ

ル・データベースの主要なスキーマ・バージョンが異なるか、dbinitが未決の場合、ドッキング・クライアントはデータベースを再初期化するためにDbinitを呼び出す。

#### 【0120】

データ・マージは、S \_\_UPG \_\_COMPテーブルの「コンパイル済み」LONGカラムを更新するトランザクションを監視する。ドッキング・クライアントの場合、データ・マージは「クライアント」コンポーネントへの変更を監視する。複製エージェントの場合、データ・マージは「サーバー」コンポーネントへの変更を監視する。

#### 【0121】

データ・マージは、「監視対象」のコンポーネントについてこのカラムのトランザクションを応用するとすぐに、コミットして、トランザクションのマージを停止する。ドッキング・クライアントは、次にバージョン・チェックを行う。複製エージェントは、バージョン・チェックを行うためにサーバー・バージョン・チェッカをスタートする。要求されるバージョンの1 つまたはそれ以上が一致しない場合、ドッキング・クライアント（またはサーバー・バージョン・チェッカ）は、応用すべきアップグレード・キットを1 つまたはそれ以上探して、アップグレード・キット・アーカイブ・ファイルをダウンロードし（ローカル・ディスクにおいてアクセス可能でない場合）、アップグレードに応用するためにアップグレード・ウィザードを呼び出す。

#### 【0122】

辞書キャッシュ・ファイル： 辞書キャッシュ・ファイル（例えば、サーバー・プログラムおよびドッキング・クライアントが必要とするレポジトリのサブセット）は、辞書をメモリにロードするようプログラムされた「キャッシュ」ファイルを提供する。「キャッシュ」ファイルを使用すると、辞書データを全てのリモート・データベースにロードする必要がなく、リモート・データベースのサイズを縮小し、リモート・データベースの初期化またはアップグレードの時間も短縮する。「キャッシュ」ファイルに既存のdiccache.datメカニズムを使用することが可能であり、本発明の範囲に属する。diccache.datファイルは、G 辞書を表

すファイルを含んでいる。動作中、バージョンを識別するためにバージョン番号がdiccache.datファイルに記憶され、一般には< \_\_ROOT>;/bin ディレクトリにあるdiccache.datファイルからバージョン番号を読み取るための新しい方法を追加する。

#### 【0123】

本発明について望ましい実施態様および実施例に関連して説明したが、それにより本発明の範囲を制限するものではない。

#### 【図面の簡単な説明】

##### 【図1】

図1 は、ソフトウェア・アップグレードの定義のプロセスを示している。図に示されるとおり、管理者は、アップグレード・キットを定義し、それによりアップグレード・キット・ファイルを生成する。次に、クライアントは、データベース・サーバーに常駐するサーバー・データベースにアップグレード・キット定義を書き込み、さらにアップグレード・キット・ファイルを生成する。クライアントは、ファイル・サーバーにもアップグレード・キット・ファイルをコピーする。アップグレード・キット・ファイルがリリースされると、アップグレード・キット・アーカイブ・ファイルが作成される。

##### 【図2】

図2 は、アップグレードの開始および分配を示している。図2 に示されるプロセスにおいて、管理者は、ソフトウェア・アイテムについて要求されるバージョンを更新する。これにより、スタートアップされると、データベース・サーバーはソフトウェアの要求されるバージョンを各ソフトウェア・アイテムごとにローカルにインストールされているバージョンと比較する。アップグレードが必要とされる場合、クライアントはファイル・サーバーからアップグレード・キット・アーカイブ・ファイルを検索し、ローカルにアップグレードを応用するためにアップグレード・ウィザードを呼び出す。移動ユーザーのアップグレードについても図2 に示されている。移動ユーザーの場合、アプリケーション・サーバーが、要求される新しいバージョンを移動ユーザーおよび地域ユーザーに送る。ドッキング・セッション後、ドッキング・クライアントは、要求されるバージョンをロ

ーカルにインストールされているバージョンと比較する。アップグレードが必要とされる場合、ドッキング・クライアントはファイル・サーバーからアップグレード・キット・アーカイブ・ファイルを検索して、ローカル・マシンにアップグレードを応用するためにアップグレード・ウィザードを呼び出す。

【図3】

図3 は、カスタマのためのアップグレード手順を示している。図に示されるとおり、アップグレード分配CDまたはディスクットが作成され、カスタマに出荷される。カスタマはテスト環境をアップグレードし、ソフトウェア・テストの新しいバージョンを分配して、ソフトウェアをテストする。テストが合格であれば、カスタマは生産ユーザーに新しいバージョンを分配する。

【図4】

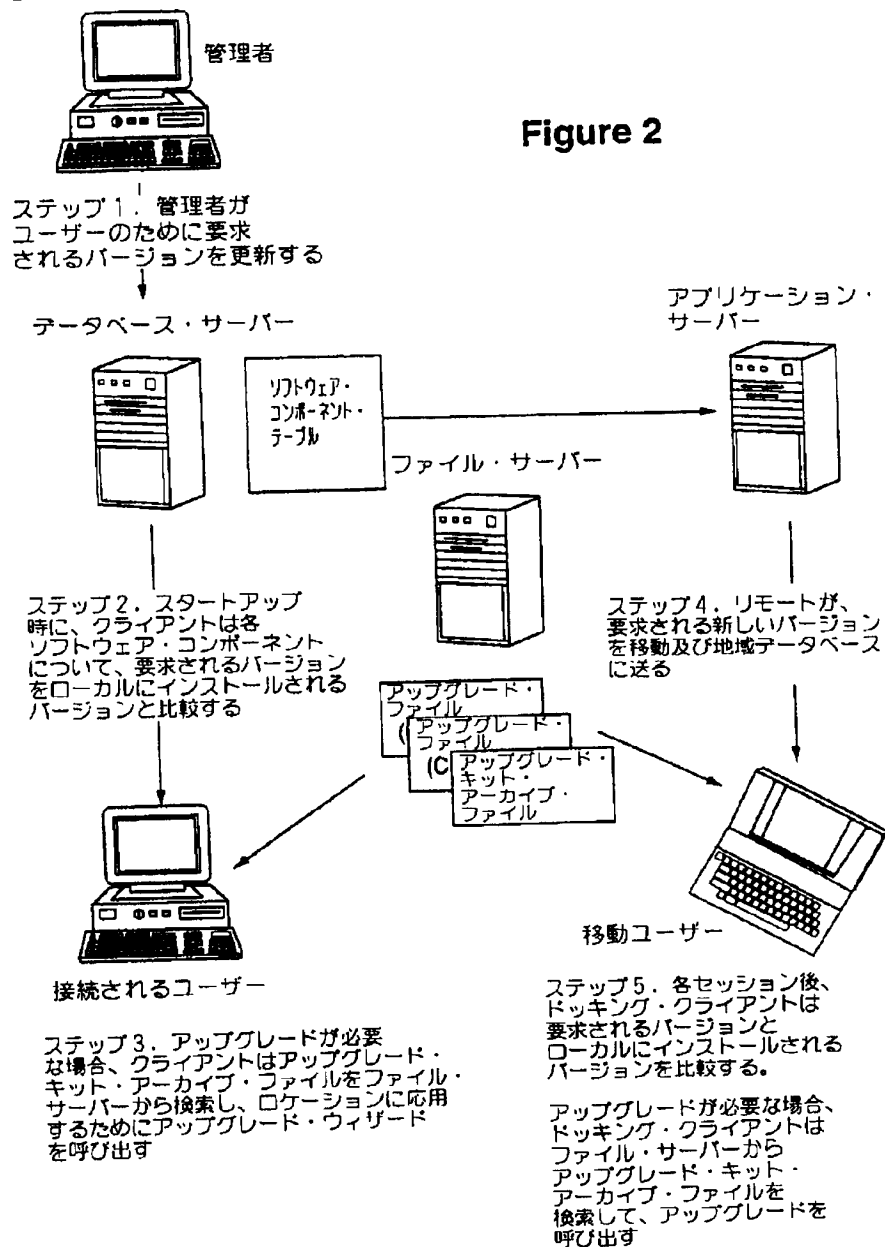
図4 は、リモート・アップグレードのためのデータ・モデルである。この図は、アップグレード・キット、コンポーネントおよび関連コンポーネントを含むキット・コンポーネント、キット・アイテムおよびキット・アイテム引数、およびキット・アイテム・タイプ引数を示す。

【図5】

表1は、ディレクトリへのファイルのコピー、ディレクトリからのファイルの削除、スタンドアロン実行可能ファイルの呼び出し、SQL ファイルのステートメントの実行、データのロード、ddl の発行、レポジトリ・ファイルのロードなど、アップグレード・キット・アイテムのための有効動作が記載されている。



【図2】



【図3】

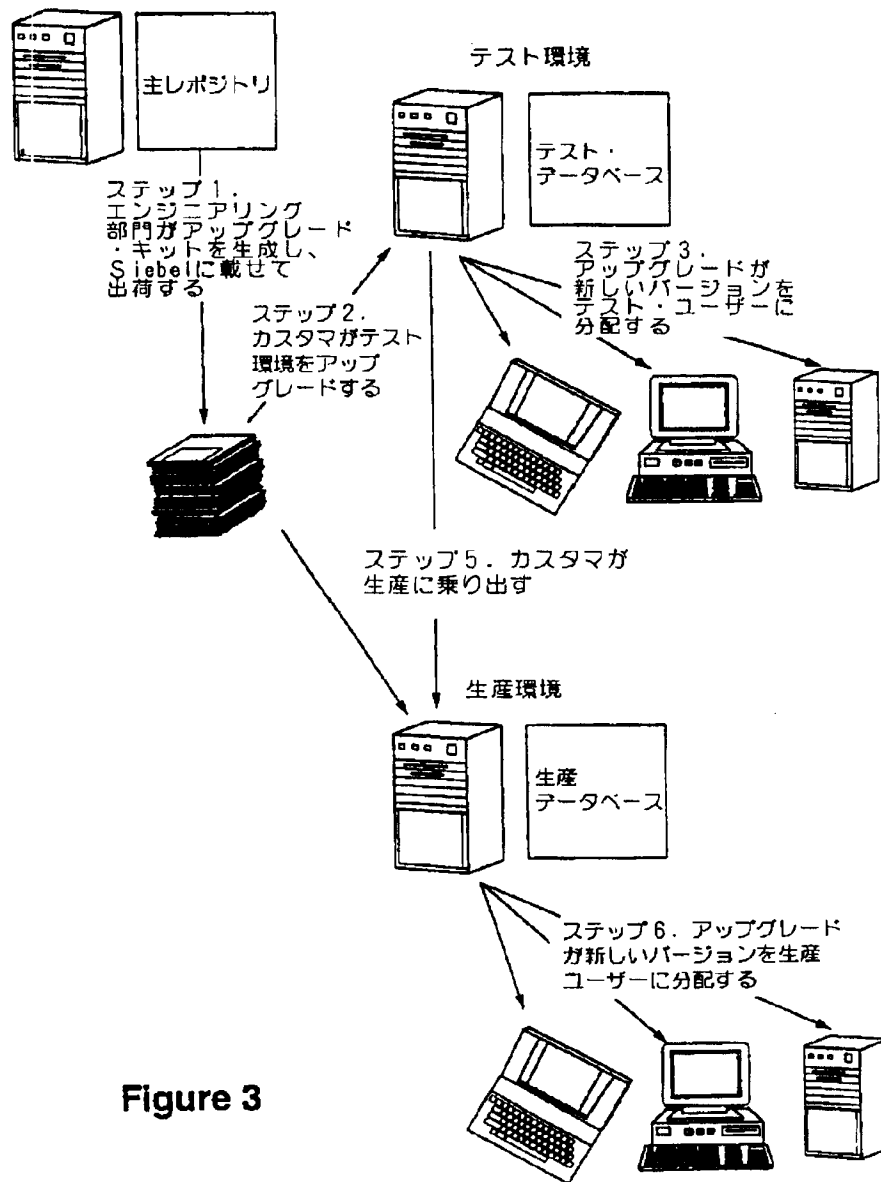


Figure 3



【図4】

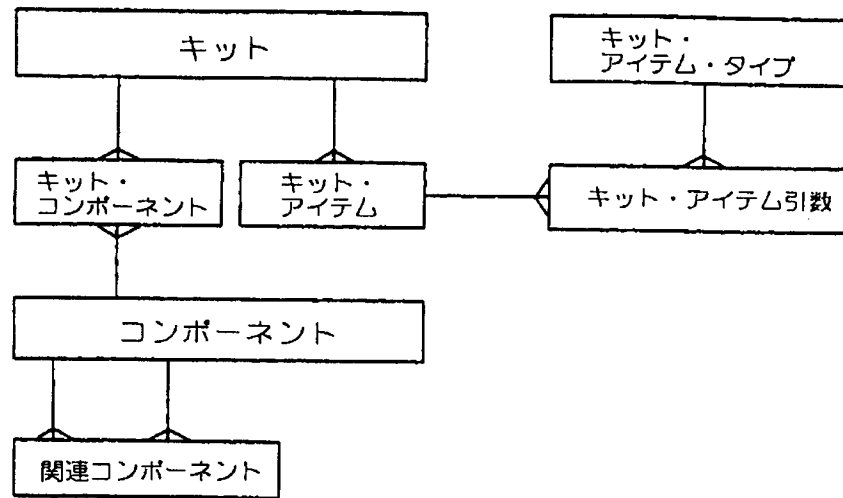


Figure 4

【図5】

| 動作          | 引数              | 説明                         |
|-------------|-----------------|----------------------------|
| FileCopy    | File, directory | ディレクトリにファイルをコピーする          |
| FilePatch   | File, directory | コピーと同様、ただしインクリメント式データ技術を使う |
| FileDelete  | File, directory | ディレクトリからファイルを削除する          |
| FileExecute | ExeFile,        | スタンドアローン実行可能ファイル（対話式の場合    |
| e           | exeArgs         | がある）を呼び出す                  |
| DbSql       | SqlFile         | sqlFileの中のステートメントを実行する     |
| DbDAT       | DataFile        | dataFileからデータをロードする        |
| DbDDL       | DDLFile         | ローカル・データベース・スキーマをファイル      |
| Repimexp    | RepimexpFile    | と一致させるために必要なddlを発行する       |
|             |                 | repimexpFileからレポジトリ・データを   |
|             |                 | ロードする                      |

Figure 5

【手続補正書】特許協力条約第34条補正の翻訳文提出書

【提出日】平成12年3月7日(2000. 3. 7)

【手続補正1】

【補正対象書類名】明細書

【補正対象項目名】特許請求の範囲

【補正方法】変更

【補正内容】

【特許請求の範囲】

【請求項1】 クライアント・サーバー・システムにおけるクライアント・コンピュータ上にソフトウェア・アップグレードを分配し及びインストールする方法であって、

クライアントに関してインストールされたソフトウェアの最小バージョン及び最大バージョンを設定することと、

該インストールされたソフトウェアの最小バージョンおよび最大バージョンから、1つ又はそれ以上のソフトウェア・コンポーネントを1つのバージョンから別のバージョンにアップグレードするためのファイル、動作およびアップグレード・ウィザードからなるソフトウェア・バージョン・アップグレード・キットの内容を定義することと、

アップグレード・キットの内容及び該当のアップグレードがどのソフトウェア・コンポーネントを必要とするかを示すテーブル・オブ・コンテンツとしてソフトウェア・バージョン・アップグレード・キットの内容をデータベースに書き込むことと、

該クライアント・コンピュータ上のアプリケーションをインスタンス化することと、

現在走行しているプログラムに関するソフトウェア・コンポーネントが最新であるかを確認することと、

現在走行しているプログラムにおけるいずれかのソフトウェア・コンポーネントが最新でない場合に、該クライアント上の他のインストールされたソフトウェアに質問して該クライアント上の該インストールされたバージョンを判定するこ

とと、

ソフトウェア・バージョン・アップグレード・キットのテーブル・オブ・コンテンツをクライアント上にインストールされたソフトウェアと比較して、現在インストールされたソフトウェア・バージョンから要求されるソフトウェア・バージョンにアップグレードするためにクライアント上に呼び出される必要のあるソフトウェア・バージョン・アップグレード・キットを判定することと、

該テーブル・オブ・コンテンツから該ソフトウェア・アップグレード・キットを構成することと、

該サーバーからアップグレードされるべきクライアントに該ソフトウェア・アップグレード・キットのコピーをダウンロードすることと、

該アップグレードを適用する前にローカル・ファイルのバックアップを作成することと、

ソフトウェア・バージョン・アップグレードのために実行されるべきアップグレード・キット・アイテムのリストをドライバ・ファイルから読み取るとともに実行されるべきアップグレードのリスト及び各アップグレードの状態を状態ファイルから読み取るアップグレード・ウィザードを、クライアント上のソフトウェアをアップグレードするために呼び出し、及び、該アップグレードを実行することと、

該クライアント上のソフトウェアをアップグレード後、該アップグレードを呼び出した元のプログラムをリスタートすることと、

を具備する方法。

【請求項2】 アップグレードの進行を追跡し、エラーが生じた場合変更をロールバックし、バックアップされたローカル・ファイルをインスタンス化し、及びセーブ・ポイントからアップグレードをリスタートする、請求項1に記載の方法。

【請求項3】 サーバーからクライアント上のソフトウェア・アップグレードの進行を監視することを具備する、請求項1に記載の方法。

【請求項4】 クライアントがアップグレードを要求する前に、1つ又はそれ以上のアップグレード・キットをクライアントにダウンロードする、請求項1

に記載の方法。

【請求項5】 アップグレードされるソフトウェアがデータベース管理ソフトウェアであり、アップグレードがさらにデータベース・スキーマ変更を含む、請求項1に記載の方法。

【請求項6】 サーバー・コンピュータ及び該サーバー・コンピュータと周期的にドッキングする少なくとも1つのクライアント・コンピュータを具備するクライアント・サーバー・コンピュータ・システムであって、前記サーバー・コンピュータはデータベース管理システム及びその上のデータベースを含み、前記データベースはドッキング中にサーバー・コンピュータからクライアント・コンピュータに部分的に複製され、前記クライアント・サーバー・システムはサーバー・コンピュータからクライアント・コンピュータ上に常駐するソフトウェアにアップグレードを適用するように構成されており、前記クライアント・サーバー・コンピュータ・システムは、更に、

クライアント・コンピュータに関してインストールされたソフトウェアの最小バージョン及び最大バージョンを設定し、

該インストールされたソフトウェアの最小バージョンおよび最大バージョンから、1つ又はそれ以上のソフトウェア・コンポーネントを1つのバージョンから別のバージョンにアップグレードするためのファイル、動作およびアップグレード・ウィザードからなるソフトウェア・バージョン・アップグレード・キットの内容を定義し、

アップグレード・キットの内容及び該当のアップグレードがどのソフトウェア・コンポーネントを必要とするかを示すテーブル・オブ・コンテンツとしてソフトウェア・バージョン・アップグレード・キットの内容をデータベースに書き込み、

該クライアント上のソフトウェア・プログラムをインスタンス化し、

現在走行しているプログラムに関するソフトウェア・コンポーネントが最新であるかを確認し、

現在走行しているプログラムにおけるいずれかのソフトウェア・コンポーネントが最新でない場合に、該クライアント上の他のインストールされたソフトウェ

アに質問して該クライアント上の該インストールされたバージョンを判定し、

ソフトウェア・バージョン・アップグレード・キットのテーブル・オブ・コンテンツをクライアント上にインストールされたソフトウェアと比較して、現在インストールされたソフトウェア・バージョンから要求されるソフトウェア・バージョンにアップグレードするためにクライアント上に呼び出される必要のあるソフトウェア・バージョン・アップグレード・キットを判定し、

該テーブル・オブ・コンテンツから該ソフトウェア・アップグレード・キットを構成し、

該サーバーからアップグレードされるべきクライアントに該ソフトウェア・アップグレード・キットのコピーをダウンロードし、

該アップグレードを適用する前にローカル・ファイルのバックアップを該クライアント・コンピュータ上に作成し、

ソフトウェア・バージョン・アップグレードのために実行されるべきアップグレード・キット・アイテムのリストをドライバ・ファイルから読み取るとともに実行されるべきアップグレードのリスト及び各アップグレードの状態を状態ファイルから読み取るアップグレード・ウィザードを、クライアント上のソフトウェアをアップグレードするために呼び出し、及び、該アップグレードを実行し、

該アップグレードを呼び出した元のソフトウェア・プログラムをリスタートする、

ように構成されているクライアント・サーバー・コンピュータ・システム。

【請求項7】 アップグレードの進行を追跡し、エラーが生じた場合変更をロールバックし、バックアップされたローカル・ファイルをインスタンス化し、及びセーブ・ポイントからアップグレードをリスタートするように構成されている、請求項6に記載のクライアント・サーバー・コンピュータ・システム。

【請求項8】 サーバーからクライアント上のソフトウェア・アップグレードの進行を監視するように構成されている、請求項6に記載のクライアント・サーバー・コンピュータ・システム。

【請求項9】 ドッキング中クライアントがアップグレードを要求する前に1つ又はそれ以上のアップグレード・キットをクライアントにダウンロードし

ッキング後に該アップグレードをインストールするように構成されている、請求項6に記載のクライアント・サーバー・コンピュータ・システム。

【請求項10】 サーバー・コンピュータからクライアント・コンピュータ上に常駐するソフトウェアにアップグレードを適用し、該クライアント・コンピュータ上に常駐するデータベース・スキーマをアップグレードするように構成されている、請求項6に記載のクライアント・サーバー・コンピュータ・システム。

## 【国際調査報告】

| INTERNATIONAL SEARCH REPORT   |   | International application No.<br>PCT/US99/04381  |
|---|---|--|
| <b>A. CLASSIFICATION OF SUBJECT MATTER</b><br>IFC(6) : G06F 09/445<br>US CL : 395/712<br>According to International Patent Classification (IPC) or to both national classification and IPC  |   |  |
| <b>B. FIELDS SEARCHED</b><br>Minimum documentation searched (classification system followed by classification symbols)<br>U.S. : 395/712; 707/203<br>Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched  |   |  |
| Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)<br>DEIALOG, DERWENT<br>search terms: software upgrade, database schema upgrades, upgrade kit, database synchronization   |   |  |
| <b>C. DOCUMENTS CONSIDERED TO BE RELEVANT</b>   |   |  |
| Category*   | Citation of document, with indication, where appropriate, of the relevant passages  | Relevant to claim No.  |
| X<br>---<br>Y   | EP 0 811 942 A2 (CHENG et al.) 10 December 1997, pages 5-18, Figures 1, 2, 9, 10, 13 and 17   | 1, 4-5, 9-17, 22-29, 31-32, 39-42, 45-46, 50, 51<br>2-3, 6-8, 33-38, 49, 52                  |
| Y   | US 5,715,462 A (IWAMOTO et al.) 03 February 1998, col. 5, lines 18-55, col. 4, lines 59-61, Figures 1 and 3-4                                 | 2-3, 7-8, 33-36  |
| Y   | M2 Communications, Siebel Announces Production Shipment of Siebel Enterprise Applications Version 3.0, M2 Presswire, 28 February 1997, page 2 | 6-8, 37, 38, 49, 52  |
| <input checked="" type="checkbox"/> Further documents are listed in the continuation of Box C. <input type="checkbox"/> See patent family annex.  |   |  |
| * Special categories of cited documents:<br>"A" document defining the general state of the art which is not considered to be of particular relevance<br>"B" earlier document published on or after the international filing date<br>"L" document which may throw doubt on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)<br>"O" document referring to an oral disclosure, use, exhibition or other means<br>"P" document published prior to the international filing date but later than the priority date claimed<br>"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention<br>"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone<br>"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combinations being obvious to a person skilled in the art<br>"Z" document member of the same patent family |   |  |
| Date of the actual completion of the international search<br>10 MAY 1999  |   | Date of mailing of the international search report<br>26 MAY 1999                            |
| Name and mailing address of the ISA/US<br>Commissioner of Patents and Trademarks<br>Box PCT<br>Washington, D.C. 20231<br>Facsimile No. (703) 305-3230   |   | Authorized officer<br>BRIAN SATTIZAHN <i>Brian Sattizahn</i><br>Telephone No. (703) 305-3800 |



## INTERNATIONAL SEARCH REPORT

International application No.  
PCT/US99/04581

| C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT |  |                       |
|---|--|-----------------------|
| Category*   | Citation of document, with indication, where appropriate, of the relevant passages | Relevant to claim No. |
| A   | US 5,155,847 A (KIROUAC et al.) 13 October 1992, see entire document               |                       |
| A   | US 5,586,304 (STUPEK, JR. et al.) 17 December 1996, see entire document            |                       |
| A   | US 5,666,501 A (JONES et al.) 09 September 1997, see entire document               |                       |
| A   | US 5,684,952 A (STEIN) 04 November 1997, see entire document                       |                       |

---

 フロントページの続き

(81)指定国 EP(AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OA(BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG), AP(GH, GM, KE, LS, MW, SD, SL, SZ, UG, ZW), EA(AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, US, UZ, VN, YU, ZW

(72)発明者 ブロダーセン, ロバート エー.  
アメリカ合衆国, カリフォルニア 94065,  
レッドウッド シティ, スピネイカーブ  
ース 17

(72)発明者 チャッタージー, プラシャント  
アメリカ合衆国, カリフォルニア 95070,  
サリトガ, グレンモント ドライブ  
21231

(72)発明者 リム, ピーター エス.  
アメリカ合衆国, カリフォルニア 94065,  
レッドウッド シティ, ガバナーズ ベイ  
ドライブ 917

(72)発明者 ワンドレス, デュエイン  
アメリカ合衆国, カリフォルニア 94062,  
レッドウッド シティ, エッジウッド ロ  
ード 1259

Fターム(参考) 5B076 AC01 AC03 AC05 EA17 EA18